LinkIt: Design, development and testing of a semi-quantitative computer modelling tool

by

FÁBIO FERRENTINI SAMPAIO

Thesis submitted in part fulfilment of the requirements of the Ph.D. degree

Department of Science and Technology
Institute of Education,
University of London.

October, 1996

ABSTRACT

This research is about the design, development and testing of a semi-quantitative computer modelling tool called LinkIt.

The aim of the software is to provide secondary students with a computational environment where they can think at a system level about models and the modelling process by expressing and testing their own ideas about phenomena without having to pay attention to the analytical relations between the variables involved.

The research involved two exploratory studies using two different versions of the software. These studies were carried out in Rio de Janeiro - Brazil with students aged 13-18 years old. During the studies, the students worked in pairs and used the computer tool to perform the expressive and exploratory tasks presented to them. The interviews were tape-recorded, some were also video-recorded and the models used and created by the students together with the steps to create them were saved and used for later analysis.

The design of the first version of the software - LinkIt I - was based on an analysis of another computer modelling system called IQON. This first version of the software was then tested with students during a Preliminary study.

The analysis of the data collected led to a rethinking of the conceptual model of the system. A new interface and changes in the properties of the objects of the system were discussed and implemented, resulting in a new version of the software: LinkIt II .

The second (Core) study aimed to investigate students' success and failure with the new version of the system, paying attention to the ease of use and learnability of the software, as well as to how they explored and externalised their ideas when using it. The analysis of this study provided evidence that the students could learn to use the software in a short period of time and that they could undertake work of at least potential educational value with the tool.

ACKNOWLEDGEMENTS

I am finishing this work with a strong belief that working with models can help us a lot to understand the world around us. However, in some circumstances, interacting with the "real thing" is a much better approach. During these last 4 years doing my PhD. at the Institute of Education I had the opportunity to work with Professor Jon Ogborn who is, with no doubt, a "real" supervisor (and not just a model of a supervisor). Among other important characteristics, he is a person who gathers all the necessary conditions to introduce students into the research realm and to make them think deeply about it. During this time he has been continually available to me, offering advice, suggestion and encouragement. For all of this I am deeply grateful to him.

We cannot pretend do a PhD. without undergoing a deep reflection of our "life beyond the PhD.". Doing a PhD. influences a lot the way we live and how we live influences a lot our PhD. as well. This process can well be seen as a feedback loop - although of high complexity ! - which over time alternates between positive and negative cycles of dominance. To take me out of many self reinforcing cycles that could have led me nowhere and put me back in a more well balanced feedback cycle I had to count on at least four special friends I made here in London: Isabel Martins & Joel Castro, Fani Stylianidou and Laércio Ferracioli.

May I also thank my partner Rita de Cássia. Her love, care and support has been invaluable, making me happier and more confident in my work.

There are other people that also became important persons for me: Paulo Adeodato and Sandra Leistner who shared a house and feelings with me for two years here in London; Carlos Perez, Ligia Barros and Arion Kurtz dos Santos which whom I exchanged thousands of emails where they were always supportive and informative.

I would like to thank all my colleagues who shared the research room with me as well as the staff of the Science and Technology Department. A special thank goes to Angela Kight and Janet Maxwell at the Technicians room.

I am grateful to all my colleagues of the Núcleo de Computação Eletrônica at Universidade Federal do Rio de Janeiro - Brazil, for permitting my absence over the last four years and also to CNPq for its financial support.

In addition I would like to thank the schools that helped me to carry out my field work: Colégio de Aplicação da U.F.R.J., Centro Educacional de Niterói, Colégio São Paulo and Escola Tenente Antonio João.

And, finally, strong thanks to my family (specially my parents and grandparents) for all of their support, patience and love.

CONTENTS

LIST OF APPENDICES

LIST OF TABLES

LIST OF FIGURES

LIST OF CHARTS

PART I

COMPUTATIONAL MODELLING AND THE RESEARCH

Chapter 1:

MODELS AND COMPUTATIONAL MODELLING IN EDUCATION

1.1 INTRODUCTION TO THE THESIS

The use of information technologies in schools is as diverse as the different theories that support them. However, they have a common paradigm: Information technologies somehow have the potential to augment the cognitive faculties of the learners (Chen, 1994). The term "cognitive tool" is thus used to refer to tools that have some cognitive properties or tools that are usefully employed in the performance of cognitive tasks (Kozma, 1987; Nickerson, 1988).

This thesis is concerned with the second of these two possibilities. In particular, it is about the design, development and testing of a computer modelling system to help students to externalise their thoughts and to think about them.

1.2 THE APPROACH

This work consists of four main parts. The first part (Chapters 1 to 3) presents modelling systems, discusses some research using these systems in educational settings and discusses the rationale for the present research. The second part (Chapters 4 and 5) is about the design, implementation and Preliminary study of a first version of a modelling tool, LinkIt (Prototype I). The third part (Chapters 6 to 10) considers, on the basis of the Preliminary study, the design and implementation of a second version of the software (Prototype II) and presents a Core study in which the tool was used. This gives an account of students' ability to manage the software and to express/ explore their knowledge in a number of domains. The fourth and last part (Chapter 11) summarises the findings of the research and presents some ideas for further research.

1.3 MODELS AND MODELLING SYSTEMS

A model is a new 'world' that someone constructs to represent things from our world or from an imaginary one. Generally such models are simpler than the world they represent and we can work (or interact) with them to understand how things function both in the model world and, perhaps, in the modelled world as well.

Another important aspect of models and the process of modelling is that the same reality can be modelled in different ways, representing different aspects of the problem or different views of the modeller. A model of the economy of a country and its implications for inflation can, for example, be very different from different political perspectives. But here what is most important is the possibility the model gives to discuss ideas about a certain problem.

The construction and use of models is not something restricted to scientific environments. Since the beginning of our lives we are accustomed to working with models. For example, when a child is asked to draw a house on a piece of paper, what she/he produces is a model of what she/he thinks is a house. Maybe the house is not very well scaled (it could have windows bigger than doors, trees smaller than people) or not all the details are included, but the most important thing is that it represents some essential aspects of the child's point of view.

The word *model* has different meanings for different areas of knowledge such as logic, engineering and common speech. A model of an axiom system, for instance, is a data structure in which those axioms are valid and can be seen as a set of laws that govern a certain world. But when someone is talking about a model of an aeroplane he/she is probably interested in a simplified system that simulates some significant characteristics of another system that, in this case, belongs to the real world. In its turn, a model of a bird (or a prototypical bird) is an ideal or standard one that can be used to make comparisons with or identify other animals.

Each of these examples captures a different aspect of modelling systems: they permit the representation of significant structures and events of a certain world; they have a set of rules that govern the functioning of their parts; and they can be used to compare/describe different representations (Sowa, 1984). Computer software that works in this way is called a computer-modelling system.

1.3.1 Simulations, Modelling and Computer Languages

At this stage it is important to differentiate modelling systems from other related applications like simulations and programming with computer languages.

A simulation is an attempt to imitate or approximate something imaginary or in the real world. If we are thinking about computer simulations we can see them as a piece of software that tries to mimic the behaviour of a certain domain. According to Steed (1992) the difference between models and simulations is that " (models are) *a representation of structures whereas a simulation infers a process of interaction between the structures of the model to create a behaviour* ". In other words, simulations pay attention to the results (output) given by the execution of the (hidden) model they contain.

In its essence a computer modelling tool is a computer language. What mainly differentiates one from the other is the level of granularity of their primitives. In conventional computer languages the primitives are more general (and elementary) - so as to permit the exploration of many fields - while requiring more programming skills of users to develop their own models. However, as the problems to be presented in modelling systems are more specific, the set of conceptual primitives existing in these systems can be more powerful (though less general), requiring less programming skills of its users.

A modelling system could be used to create both models and simulations. For instance, you could use a modelling system like STELLA (Steed, 1992) to create a model to represent the interrelations among the employees of an industrial company and permit people who are interested in increasing the profits of the company, to try some simulations with that model, changing some its parameters (e.g. salary of the employees).

With the advent of direct-manipulation interfaces and scripting languages, the differences between these three modalities are becoming less and less "visible". Examples are the Hypercard and Excel environments where it is easy to construct, modify/adapt and test models about a certain domain under certain conditions.

1.4 MODELLING SYSTEMS: A CLASSIFICATION

There are many different characteristics of modelling systems that could be used to classify them. As this study is mainly concerned with models that can be implemented computationally and focuses on aspects related to their use in educational environments, I will use a classification devised by Bliss & Ogborn (1992c) that focuses on the way to express the ideas to be modelled (For more information see (Gilbert & Osborne, 1980 ; Ogborn, 1993).

1.4.1 Quantitative Models

Quantitative models are strongly based on variables and mathematical relations between them to describe (or to model) a situation in the world. So to describe a situation, the user must be able to identify its variables and specify the exact functional relation between them. From this perspective, if someone wants to explain how the velocity of car is changing over a certain period of time, it would be necessary to have an equation system like this:

$V := V + a * DT$
$X := X + V * DT$
$a := ?$
$DT := ?$
$V_{initial} := ?$
$X_{initial} := ?$

Quantitative computational modelling systems are probably the best known kind of modelling system, with applications ranging from daily-life activities to scientific work. However, this formal analytic approach to represent ideas does not permit one to have deeper insight into how the behaviour represented by the solutions actually comes about (Dillon, 1994). Therefore the designers of recent computational modelling systems - especially the ones with educational purposes - have been investing in the visual representation of the models, proposing interfaces that use metaphors closely related to the problems being modelled and to the user's knowledge. The best known of such systems with an educational purpose is STELLA (Structural Thinking, Experiential Learning Laboratory with Animation) developed for Macintosh computers (the IBM PC version came later), which utilises the windows environment, the concept of object orientation and four basic building blocks to implement a fluid flow analogy (Richmond, 1987) (see Chapter 2 for further description of STELLA).

Older systems exist and are still in use. They are normally programmed in BASIC (Kurtz dos Santos, 1995) and permit the user to either enter some parameters required by the simulation or to introduce a module that describes the event to be modelled. An example of this second case is the DMS system developed for BBC computers (Wong, 1986).

An interesting case is the increasing use of spreadsheets as quantitative modelling system by teachers. Brosnan in Mellar, Bliss, Boohan, Ogborn, & Tompsett (1994: 76-77) gives two reasons for this. First, worksheets give students the possibility to focus their attention on their understanding of the underlying scientific principles of a phenomenon, leaving to the software the (almost total) control of the "mechanical mathematics" that governs the phenomenon, and second, they can be used in many different conceptual areas across the curriculum.

1.4.2 Qualitative Models

Qualitative models are based on a descriptive specification of the objects and their relations in the world to be modelled. In our daily life we are familiar with this mechanism for explaining to someone else how things function. Although such qualitative models are not very suitable in applications where you want to repeat or simulate situations in an automatic way, some computational modelling systems like Linx88 (Briggs, Nichol, Brough, & Dean, 1989), VARILAB (Hartley, Byard, & Mallen, 1991) and 'Explore your Options' (Bliss & Ogborn, 1992a), permit such descriptions to be made using a graph metaphor, with nodes (objects) and links (relationship between objects), so that some automation can be achieved.

1.4.3 Semi-quantitative Models

These models are based on descriptions that use ordinal (rather than numerical) relationships such as "X increases Y" or "X decreases Y", to describe a certain behaviour and are also very much used to describe daily-life situations such as "if the number of customers in a supermarket increases than the queue at the checkout increases... if we increase the number of checkouts the queue will decrease". Although relationships such as 'increases', 'decreases', 'greater', 'less' and so on are not mathematically precise, they serve to externalise important information about how and why changes occur.

It might be thought that this kind of construction is only used by naive or 'plain folks' to explain the behaviour of certain situations, but researchers in the artificial intelligence and cognitive science communities have argued that semi-quantitative reasoning also seems to be used by experts to express causal explanations of the behaviour of physical systems (de Kleer & Brown, 1983; Dillon, 1994; Kuipers, 1994). One possible consequence of this argument is that semiquantitative modelling systems can contribute to a deeper understanding about the world by permitting people to externalise and discuss their ideas about a certain domain, focusing on the conceptual level of the problem instead of its functional level.

1.4.4 Dynamic versus Static Models

Another important dimension of modelling systems is their relation with time. Modelling systems that permit the construction of models that change over time are called dynamic modelling systems. Otherwise they are static modelling systems . So a modelling system that permits the construction of dynamic models has to have one module that permits the modeller to express the concepts and ideas to be modelled (representation module) and another that is responsible for calculating the evolution of the model over time (processing module), giving feedback to the modeller. However, static modelling systems may not need a processing module. A good example is a hypertext system used to represent someone's ideas about the most interesting tourist places in London. Static modelling systems that do have a processing module are mainly concerned with the satisfaction of constraints. An example is a spreadsheet used to calculate the impact on the profit of a company if you raise the salary of its employees.

One important aspect of dynamic modelling systems is that they serve well to represent (and therefore to help to think about) real life situations. They are at the basis of the system dynamics framework (Forrester, 1968) which tries to understand certain phenomena by looking at them as a "collection of interacting elements that function together for some purpose" (Roberts, 1983) (see chapter 2 for a description of some computer modelling systems that use this approach).

1.5 COMPUTER MODELLING IN EDUCATION: WHY?

Since its beginning, scientific knowledge has been developed from questioning and observation (by "scientists" or not) of phenomena. Copernicus, through the observation of the movement of celestial bodies was led to suggest that our planet revolves around the sun; Galileo, through observation and analysis of some common daily activities suggested and developed many artefacts that helped the execution of these activities, and postulated the first rules that led to the origin of mechanics. The objective of science is to understand and explain real-world phenomena. Models are at the heart of this activity as "thinking tools" to help scientists in their tasks.

To make students "science literate" is essentially to make them think in a critical way about scientific concepts and to question them (Papert, 1980; Steed, 1992; Wong, 1993). In such an educational setting the important thing is not only to make students find the correct answers but to give them the opportunity to become active learners, engaging in a process whereby they can develop their own understanding of natural phenomena. In such an educational environment computers can be used to,

> *...explore domains where the teachers know a bit more than the students, but do not know all the answers. Domains they could model with their class, around which they could cope with their students, about which they could share some healthful moment of discovery and amazement*. (Vitale, 1988, p. 227).

The importance here therefore is not only to make students find the correct answer but to give them the opportunity to become active learners, engaging in a process where they can develop their own understanding of natural phenomena.

The best that can be done to understand the behaviour of the physical world is to guess intelligently some likely causes of that behaviour and construct mental representations of them. Following this individuals can formulate some hypothesis about the processes that are going on and predict some of their consequences. In some special circumstances, it is possible to go there (to the physical world) and test these hypotheses. These ideas of creating a model and some theories about them are, according to Gilbert & Osborne (1980: 58), "*the 'imaginative adventures of the mind' and they bring the essential ingredient of creativity into science*".

Working with physical means of representation (e.g. pencil and paper) permits the externalisation of the concepts and meanings that are embodied in mental representations, helping the learner to think about what is intended to be represented (Novak & Gowin, 1984). Using a computer modelling system to represent concepts and meanings permits the learner to go further in his/her ideas, exploring the relations between the different kinds of objects, formulating and testing hypotheses, etc. About these ideas Webb (1990) suggests:

In order to assist children in improving their mental models so that they are closer to accepted scientific theory, it is necessary for teachers to provide a learning environment which will facilitate children in reconstructing their mental models. The conditions needed for this to occur are those in which the inadequacies of a child's mental model become apparent to that child and she/he can experiment with alternative models ... (p. 6-7)

Another important characteristic of computer modelling systems is the possibility of linking multiple representations which can facilitate the process of creating meaning from representations.

The issue of representation is crucial in science and science education. Science can even be defined as a means for constructing and improving representations of the world. (Teodoro, 1992b, p. 2).

1.5.1 Barriers to Accessibility

At the present time the main problem that persists with mathematical modelling systems is the "high entry fee" that has to be paid to use them. Some authors like O'Shea & Self (1987) argue that the physical sciences are very much concerned with the development and use of mathematical models, but that their complexities are, in most cases, beyond the students' abilities. This idea suggests that work with modelling should start from a more qualitative platform. When sufficient knowledge about modelling is developed and the mathematical skills are present, students could then migrate to modelling systems which offer more quantitative results (Ogborn, 1992). About this Kurtz dos Santos (1992) says:

...it is important to begin from a secure base in observed phenomena. Much of the empirical knowledge at this stage will be qualitative, in that quantitative formulations depend on later stages being achieved, such as variables being isolated and relationships postulated. This suggest an emphasis on simple experimental work aimed at a qualitative understanding of the system: what happens and what affects what happens. (p. 56)

Another important aspect of computer modelling systems that can make them inaccessible to the new user is a user-<u>un</u>friendly interface. A common example is a system based on a command-line interface where the user has to memorise a set of rules to operate the system in advance (or at least has to have a help system available), diverting his/her attention from the cognitive-relevant aspects of the problem. Also modelling environments that incorporate too many options and a large number of expert level modelling functions can easily discourage the new user.

1.6 MOTIVATION FOR PRESENT RESEARCH

Due to the constraints of mathematical models, some authors argue that the use of semi-quantitative modelling systems can be beneficial to people without a mathematical background, e.g. young students (Bliss & Ogborn, 1992).

In considering the use of computers to implement semi-quantitative models an environment should be set up where the user is responsible for defining the variables and their relations and attaching to them some characteristics that reflect the functional aspects of the objects and events being modelled. The computer, in its turn, would be responsible for choosing the right equations to govern the evolution of the model from the characteristics of the variables and relationships already created by the user, and for making the model evolve over time.

In 1992 I became aware of research carried out by the London Mental Models Group (Bliss & Ogborn, 1992) which led to the development of a semi-quantitative modelling system called IQON (Miller & Ogborn, 1990) (see also Chapter 2). The research presented in this thesis builds on that research in the computer modelling field. A new semi-quantitative computer modelling tool - LinkIt - that is more general than IQON, but still simple to use, is proposed, designed, implemented and tested.

What follows in the next two chapters of Part I of the thesis is the presentation and discussion of some existing computer modelling systems that were influential for the design of the present system (Chapter 2) and the rationale for this research (Chapter 3).

Chapter 2:

RELATED WORK ON MODELLING SYSTEMS

2.1 INTRODUCTION

This chapter presents the main ideas of the systems dynamic approach and describes and discusses three of the most important computer modelling systems developed within this approach. The chapter concludes by relating these computer environments to the software developed as part of this research.

It is important to note that the historical account of the development of computer modelling systems in education given in Section 2.2 is a partial one, just taking into account those systems relevant to the systems dynamic approach. A broader approach concerning the use of these systems in the English educational system is presented in Mellar (1990).

2.2 SYSTEMS DYNAMIC APPROACH AND COMPUTER MODELLING FOR EDUCATION

Systems thinking - which is the concept behind the systems dynamic approach - grew rapidly at the end of the 1950s at MIT when Jay Forrester and some of his colleagues started using it to think about management problems. However the idea gained prominence in the 1960s when it was applied to urban growth and development and global patterns of the consumption of natural resources (Mandinach, 1989; Toval & Flores, 1987).

Systems thinking is a valuable tool for understanding the behaviour of complex systems over time using simpler concepts of change, causality and feedback. By specifying the rules that define the causal relationships between variables and how they change over time, it is possible to construct simple and general models that account for biological, ecological and societal problems.

In the last two decades, with the increasingly widespread use of microcomputers in schools, the systems dynamic approach has been used as the backbone of many of the computer modelling/simulation systems available for those computers (Millwood & Stevens, 1990). Most of these systems were written in BASIC and permit the students to manipulate some of their parameters. The outputs of these systems are either tables containing the values of the important variables against time or plotted graphics of these variables. Some examples of these systems can be found in Marx (1984) and Crandall (1984) who present a collection of simulation programs related to topics in mathematics, chemistry, physics and biology.

However, from an educational point of view, these computer programs have three major drawbacks: (i) they require time and computer skills for the users to enter the programs in the computer; (ii) they distract the user from engaging directly in the process of modelling; and (iii) they only permit the manipulation of some parameters, functioning more as a black box yielding little or no insight into the interrelationship of the elements of the system from which the output has arisen (Whitfield, 1988).

At least two new ideas can be mentioned as new attempts to overcome the problems presented above. The first idea is the development of computer languages for education specifically designed to construct models. One example is the language DYNAMO (an acronym for DYNAmic MOdelling) (Roberts, 1983) which has a small set of meaningful commands oriented to the mathematical description of the relationships between the variables of a problem and their output (see Figure 2.1 and 2.2).

```
*           YEAST GROWTH
L           YEAST.K = YEAST.J+(DT)(BUDDNG.JK)
N           YEAST = 10
NOTE        YEAST CELLS (CELLS)
R           BUDDNG.KL = (YEAST.K) (BUDFR)
NOTE        BUDDING (CELLS/HOUR)
C           BUDFR=0.10
NOTE        BUDDING = FRACTION (1/HOUR)
PLOT        YEAST=Y/BUDDNG=B
PRINT   YEAST,BUDDNG
SPEC    DT=1/LENGTH=30/PLTPER=1/PRTPER=5
RUN
```

Figure 2.1: A piece of program using DYNAMO language used to generate the graph shown in Figure 2.2

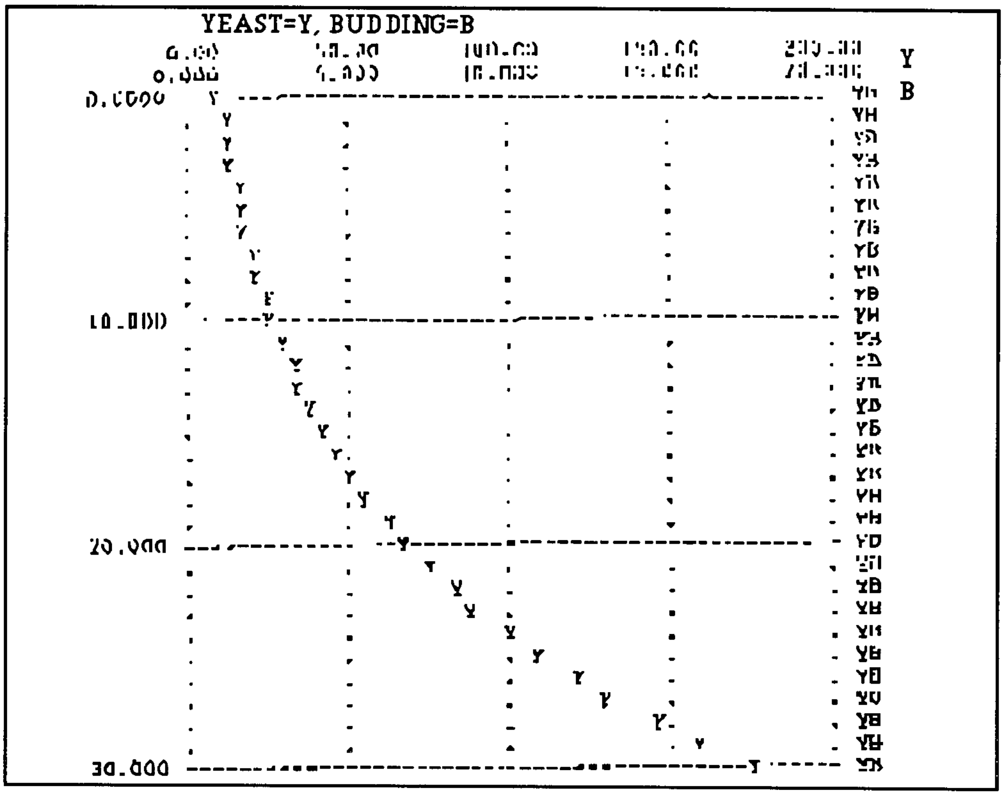


Figure 2.2: An output plot produced by DYNAMO

A second approach taken by other researchers is the development of computer dynamic modelling environments where the user just has to introduce a module that corresponds to the world to be modelled. Examples of this case are the systems MODL (Hartley, 1981), MECHANICS (Staudenmaier, 1982) and DMS (Dynamic Modelling System) (Ogborn & Wong, 1984; Wong, 1986).

It is important to note that these computer environments represent a new tendency in software systems initiated at the beginning of the 1980s: the idea of integrating different but complementary tools in order to help the user to accomplish a certain task in a shorter period of time.

DMS can be seen as a good example of this category. It was initially developed in BASIC for BBC computers (versions for Mac and RML computers came later) and integrates, in the same environment, 3 modules that are essential for the modelling process: a program editor that permits the user to interact with the system (create models, load a previous one, change parameters, etc.); a graph plotter responsible for the output of the model when it is iterating; and an empty slot where the model (actually the equations that represent it) can be inserted (see Figure 2.3 and 2.4)

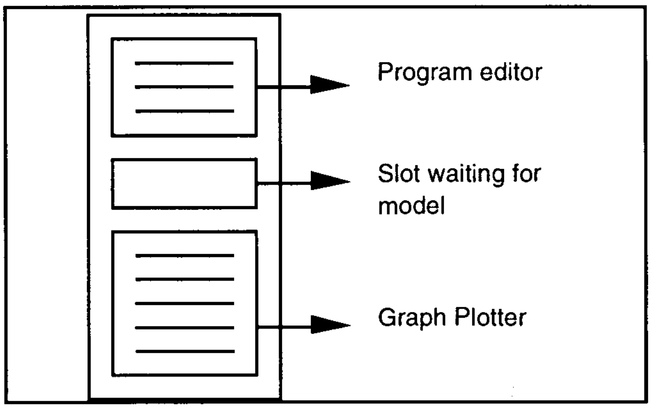


Figure 2.3: The general architecture of DMS

Although purpose-made modelling languages and computer modelling environments represented a step forward in the process of focusing the learner's attention on the process of modelling and on learning about models and classes of models, they still contain some drawbacks. First, the fact is that modelling languages, although they facilitate the process of programming, are still computer languages and therefore require the user to have computer skills to use them. Actually this idea is part of a tendency initiated in the middle of the 1970s which aimed to give the power of programming educational software to teachers by creating computer environments where they could program and test their own "computer lessons". Some of the languages very much promoted at that time were SuperPilot (Apple Computer, 1980) and TUTOR (Solomon, 1987). The basic assumption concerning the development of this kind of environment was the idea that creating purpose-made computer environments would facilitate the process of programming courseware by non-specialists. The idea did not prove fruitful for many reasons. However, there was one important lesson that was learned. It became clear that computer programming skills are not the same as just learning a computer language. They take time and practice to be acquired.

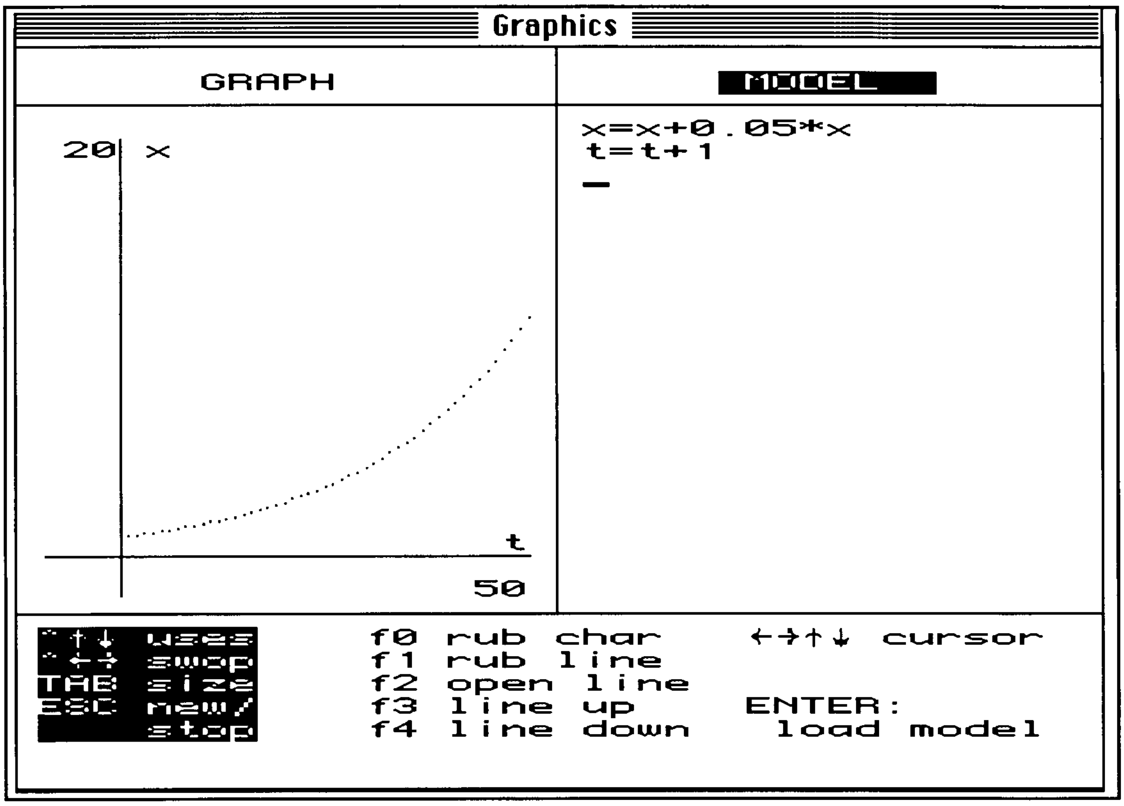


Figure 2.4: DMS screen showing a graph of a model (growing function) and the corresponding equations (right side)

A second negative aspect is that the outputs of these kinds of software are in the form of graphs and/or tables representing the values of the variables, which is not a straightforward way either to understand how the elements of the model are changing or to grasp a systemic view of the model evolving over time.

The idea of giving students full control to express and test their ideas about formal objects has become a common claim by many researchers working with computers in educational environments. However the computational metaphor of programming used so far was too basic to permit the exploration of certain fields. A new more powerful metaphor that gives students the possibility of concentrating on the objects and actions of the domain to be explored without having the burden of first having to program these domains was needed. The direct manipulation style of interface (Shneiderman, 1992) and causal loop diagrams (Roberts, 1983) provided new metaphors for a new family of computer modelling tools.

Before moving to the next section it is worth saying that the idea of "thinking with computational objects" was not something put forward in the 1980s after many years of using computer modelling software. The LOGO language developed in the beginning of the 1970s took this idea as one of its most important assumptions. Although not linked to the systems dynamic approach of working with models, LOGO offered an interesting alternative way for exploring modelling by permitting the construction of what Papert called *microworlds*

(Papert, 1980). The LOGO environment makes a great appeal of manipulating computational objects, giving the student a sense of playing with a concrete world. But rather like other computer-modelling software it needs to be programmed.

2.3 DIRECT MANIPULATION INTERFACES

The most common example of a direct manipulation interface (also referred to as a graphical object-oriented interface) is the Macintosh computer's desktop. It contains icons (pictures) representing folders, documents, discs and a wastebasket. It also contains windows which sometimes come with menu bars, scroll bars and boxes for zooming, resizing and closing (see Figure 2.5).

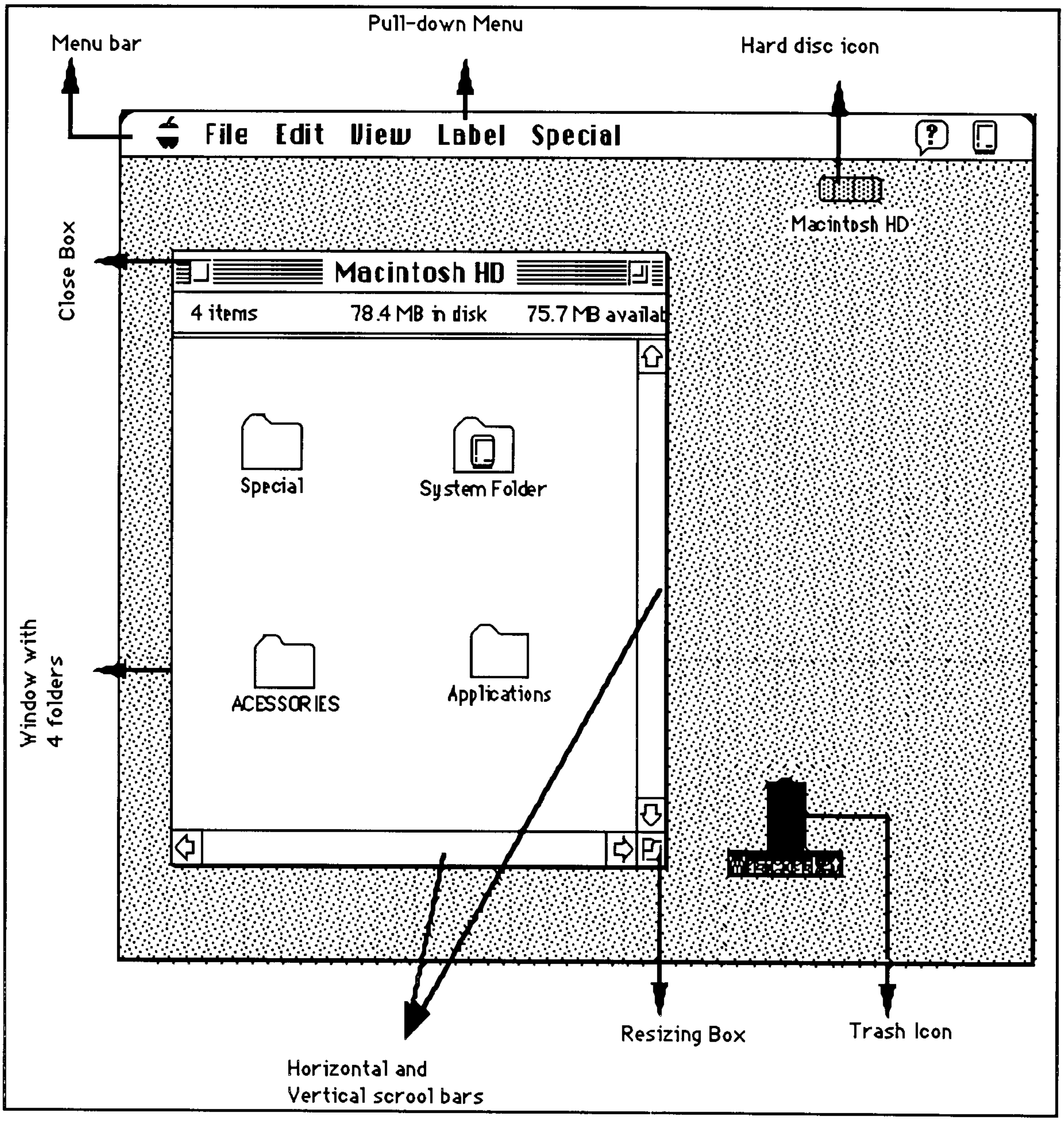


Figure 2.5: Macintosh system interface

This style of interface is also known as a WIMP interface. The term "WIMP" is an acronym for Windows, Icons, Mouse and Pull-down (or Pop-up) menus which describe the main features of this type of interface.

An important concept in the realm of direct manipulation interfaces is that of metaphor. A metaphor can be seen as an "*invisible web of terms and associations that underlies the way we speak and think about a concept* " (Erickson, 1990, p. 66). Its purpose is to help in the understanding of abstract concepts starting from more familiar and understandable knowledge[1].

The metaphor used by Macintosh computers is that of a *desktop* environment. The idea is to use the (hopefully) familiar concepts we already have about how the top of an office desk is organised to help to understand how to interact with the Macintosh system. Some familiar office objects are presented on the screen in the form of icons such as folders, files and a wastebasket. The user can "open" these objects to see and manipulate their contents.

However, it is important to notice that the desktop metaphor does not necessarily have to be the metaphor used by all graphical object-oriented systems. These two concepts have become so closely indentified in the computer interface world that people sometimes use them as synonyms. An example is the calculator existent in the same Macintosh environment mentioned above. When the calculator is evoked (normally by selecting it from the Apple menu) the system displays a drawing that resembles a pocket calculator. In order to use the computer calculator it is necessary to have knowledge of (in other words, to be familiar with) how a pocket calculator works. So, what is being used is familiar knowledge about how an electronic calculator operates, which is a different metaphor from that employed by the system.

The main advantage of direct manipulation interfaces is that their objects are dynamic and interactive. Instead of typing a line command such as "open a window and display the content of the hard disc on it", the user can directly double click on the hard disk to see its contents. So the user has a feeling that the displayed objects replace, or even become the objects and operations they represent (Clark,1993). Empirical studies have shown that after a very short period of use these objects can be manipulated without conscious attention (Smith, Irby, Kimball, & Verplank, 1982).

This style of interface is based on at least two assumptions that have important implications for the design of computer tools for education (Kozma, 1987):

• It gives the user the sense of manipulating concrete objects and therefore he/she can apply his/her knowledge of the world to interact with and understand them;

---

1 An interesting account on how we use metaphors in our daily life is given by Lakoff and Johnson (1980) in their book *Metaphors We Live By.*

• The actions upon these objects can be done in an "unconscious way", permitting a reduction in the cognitive load on the user and giving him/her the opportunity to concentrate on the work to be done. This is what Rutkowski (1982, p. 291) calls the principle of transparency: "*The user is able to apply intellect directly to the task; the tool itself seems to disappear*".

2.4 CAUSAL LOOP DIAGRAMS

Causal loop diagrams are a graphical technique used to represent causal relationships between the elements of a problem being studied (see Figure 2.6).

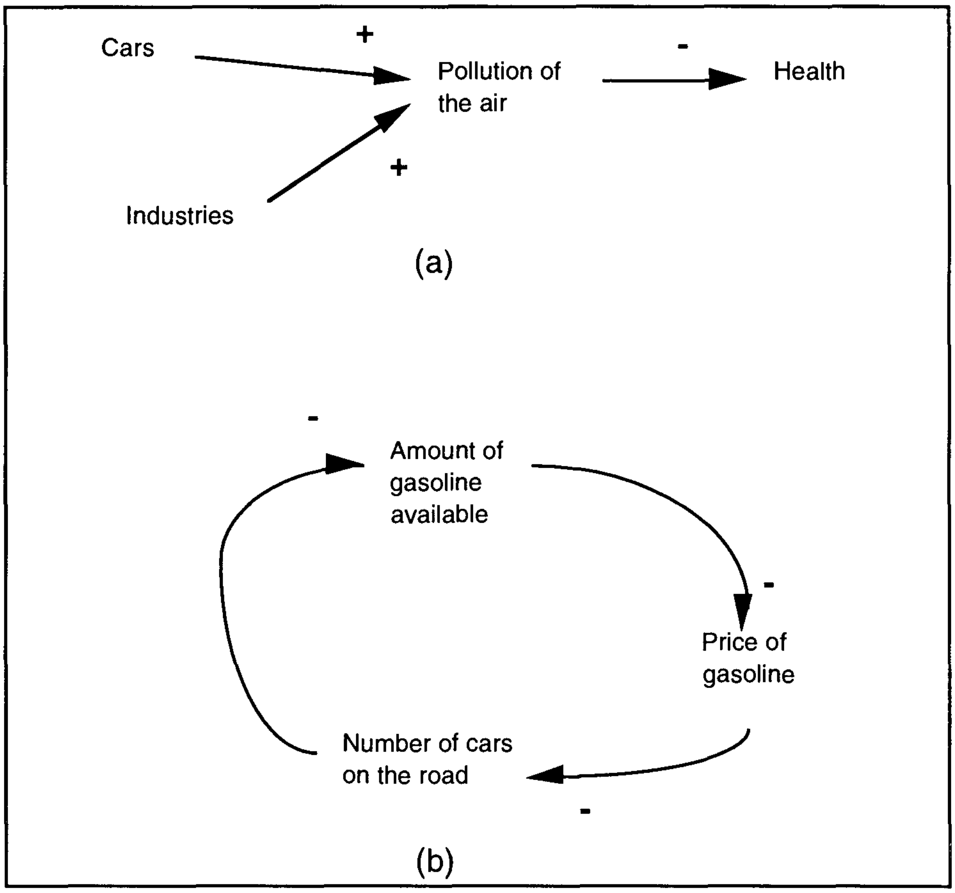


Figure 2.6: (a) Causal loop diagram with three variables (b) Causal loop diagram with three variables including a feedback loop

Basically they have three important elements:

• *Nodes* - Used to represent the elements (or variables) of the problem being modelled;

• *Arrows* or *links* - Used to represent a causal relationship between variables. The direction of the arrow indicates the direction of causation;

• *Sign of the relationship* - Indicates how a causal variable influences a dependent variable. A *positive* influence means that as time changes, the variation of the dependent variable follows the same direction as the variation of its causal factor. In that way the example shown in Figure 2.6.(a) could be read like this: " If the *number of cars* increases the *pollution of the air* will increase". The complementary idea is also implied: " If the *number of cars* decreases the *pollution of the air* will decrease".

A *negative* influence means exactly the contrary: the variation of the dependent variable goes in the contrary direction of the variation of its causal factor. In Figure 2.6.(b) the relationship between *amount of gasoline available* and *price of gasoline* could be read like this: " If the *amount of gasoline available* increases the *price of gasoline* will decrease. If the *amount of gasoline available* decreases the *price of gasoline* will increase ".

There are two other important concepts associated with causal loop diagrams: positive feedback loops and negative feedback loops. A positive feedback loop is when a causal loop diagram has a closed loop where the behavioural changes are reinforced. Over time, positive feedback loops have a tendency to present run-away growth or collapse as a consequence of the "snowball" effect (see Figure 2.7.(a)).

Negative feedback loops tend to reverse the direction of change as the loop is traversed. It is possible to see it as something resisting change or counteracting, over time, an external disturbance (see Figure 2.7.(b)).

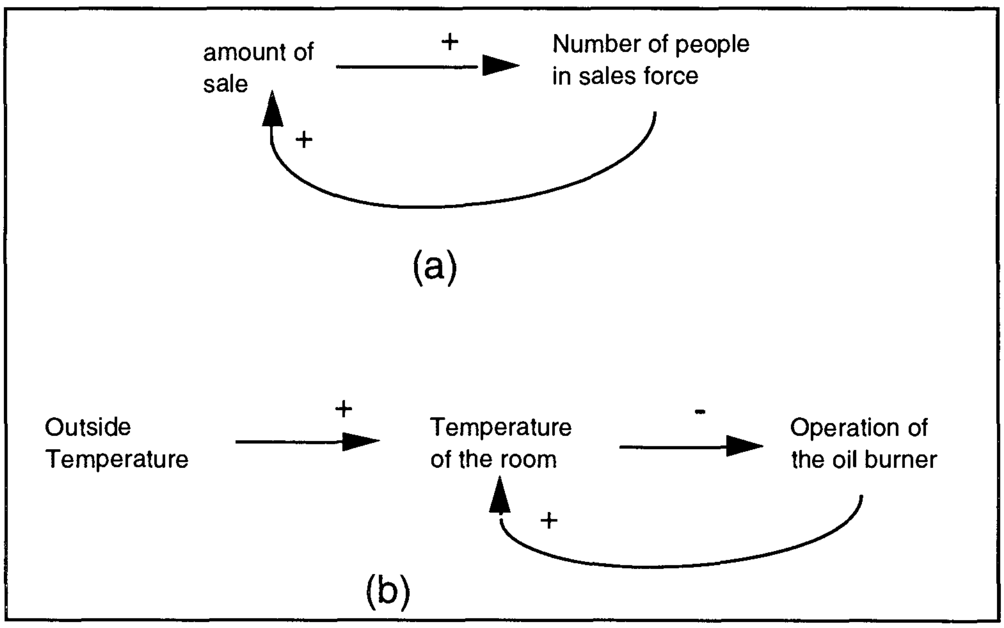


Figure 2.7: Causal loop diagrams containing feedback loops: (a) Positive feedback loop (b) Negative feedback loop

The use of causal loop diagrams provides an easy way to introduce students and teachers to dynamic modelling for the following reasons (Roberts, 1983; Toval & Flores, 1987):

- It is a graphical representation based on very simple building blocks that permits more effective communication of someone's ideas about a system;
- It permits the visualisation of the whole model facilitating the "systems view" of the problem;
- It permits a sequential approach, by stages, for constructing models.

The three computer systems - STELLA, IQON and ScienceWorks Modeler - described in the next sections are examples of how the ideas of direct manipulation interfaces and causal loop diagrams were put together to provide simple and yet powerful computer modelling tools for education.

2.5 STELLA SYSTEM

This section has two subsections. The first subsection gives a general description of STELLA (Structural Thinking Experimental Learning Laboratory with Animation) focusing on its building blocks and interface. This description is based on the writings of the following authors: Mandinach, 1989; Richmond, 1987; Steed, 1992; Whitfield, 1988.

The second subsection criticises the software assuming its use in an educational environment.

The STELLA version considered here is 2.01 for Macintoshes[2]. The system was tried both on a Power Macintosh 7100/80 and a Macintosh LC III.

2.5.1 STELLA: General Description

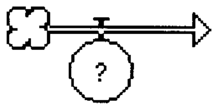
STELLA is a computer system designed for the representation of dynamic models using iconic levels, flows and converters. In a certain way it can be seen as an implementation of DYNAMO with a graphical interface.

The system takes the idea of causal loops and elaborates on them through a metaphor of tanks, pipes and flows, providing the user with four building blocks:

Stock - Represents a quantity that can increase (accumulate) or decrease (de-accumulate) over time.

---

[2] STELLA 4.0 is now available as a free Demo version. It can be dowloaded from the www site: http://rtpnet.org:80/~gotwals/stella/stella.html.



Flow - Controls the rate of incoming and outcoming material from stocks. A Flow can come from or go to a 'cloud' which means that its source or destination are not specified (it can mean that the cause or effect of the flow is not relevant to the model). Flows can be altered by stocks and converters.

Converter - Converts inputs into outputs. They can be constants or calculated from other quantities.

Connector - Connectors are used to determine that one variable in the model depends on one or more other variables in the same model.

Figure 2.8 below shows a model about inflow/outflow of a tank constructed with STELLA. There is also an inner window showing the (partially complete) mathematical equations that describe the model.

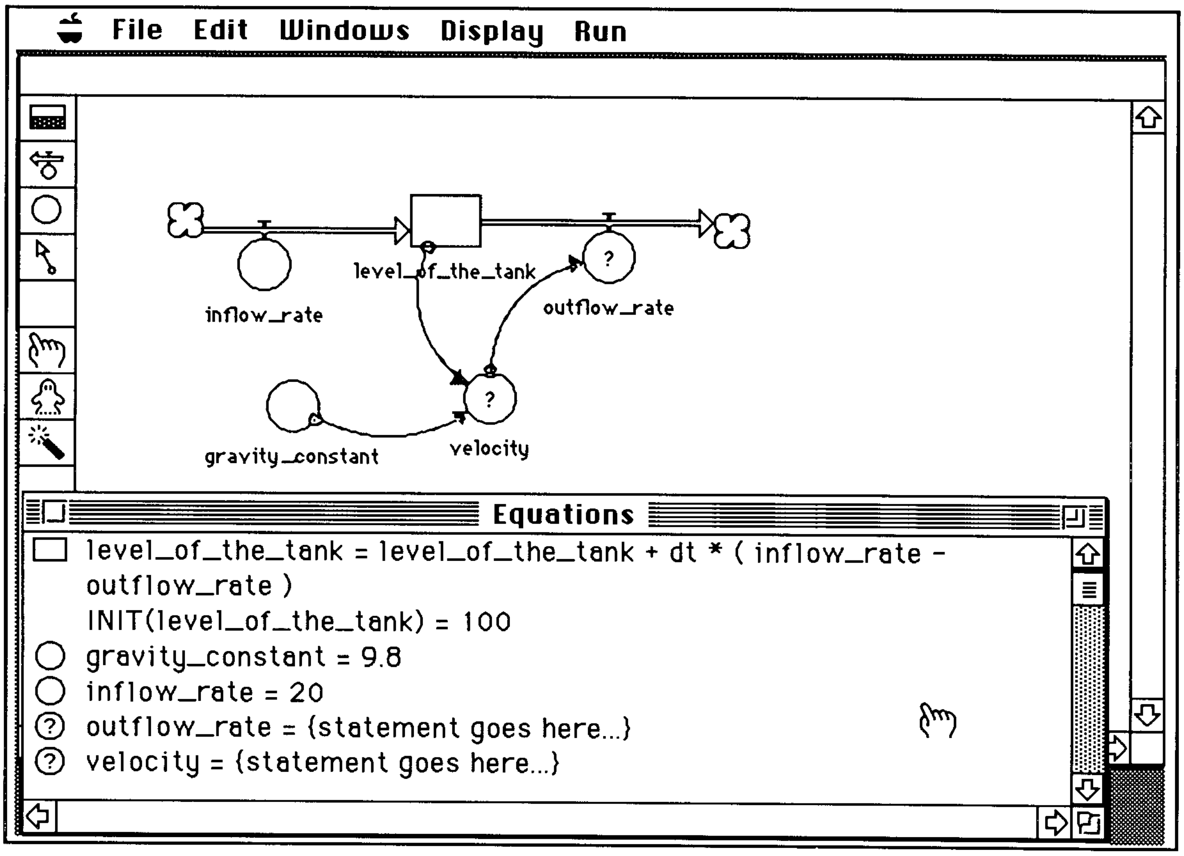


Figure 2.8: STELLA: Model about inflow/outflow of a tank with its set of equations

The control panel on the left side of the window provides the elements to edit a model. The first four icons represent the four different building blocks provided in STELLA. To create a stock or converter, the user has to select one of these objects (single click) and point and click on the place he/she wants to place the object on the working area. The last three icons correspond to the tools to manipulate the building blocks: the fifth icon (the hand) is used to select and move items; the sixth icon (the ghost) duplicates an element that can be placed anywhere on the window; the last icon, the dynamite, serves to break existing connections and variables.

After drawing the model the user has to double click on the objects on the screen to set the equations and the initial conditions of the variables. After that it is necessary to set the scales (range of variation) for the objects created and run the model (both functions are inside the *Run* pull-down menu). Another way of relating variables of the system is through a graph (drawn by the user) that specifies the relation between two variables of the model.

The outputs of STELLA can be of 3 kinds:

- Animated diagram - The graphics capabilities of the system permit the user to see in real time the flow of information in the model. The levels of the stocks move up and down representing a tank being 'filled' or 'emptied'. Flows and converters have small arrows that move across their icons as a function of their value.
- Table - Show in a table the values of the objects chosen by the user.
- Graph pad - Show a graph of the objects chosen by the user. The system can show scatter plots of two variables or a time series graph of up to four different variables.

2.5.2 Mathematics of STELLA System

After creating a diagrammatic representation of a problem on the screen, the user has to describe how the elements presented on the diagram change in time and in respect to the values of other variables and constants.

In order to do so the user has to go to each element of the model and mathematically express them in terms of their causal factors using finite algebraic relations (relations for 'stocks' are interpreted by the system as finite difference equations). At this stage the system gives some help to the user by presenting a set of elements (other variables/constants existent in the model) that are graphically connected to the variable in question. Figure 2.9 presents a window requiring the mathematical definition of the variable *outflow-rate* (shown in the model in Figure 2.8). Its only causal factor - *velocity* - appears in a small window called "Required Inputs" and can be used to describe the dynamic behaviour of *outflow-rate.* The system also provides a set of mathematical functions that can be used by the user. They appear in the window called "Builtins".

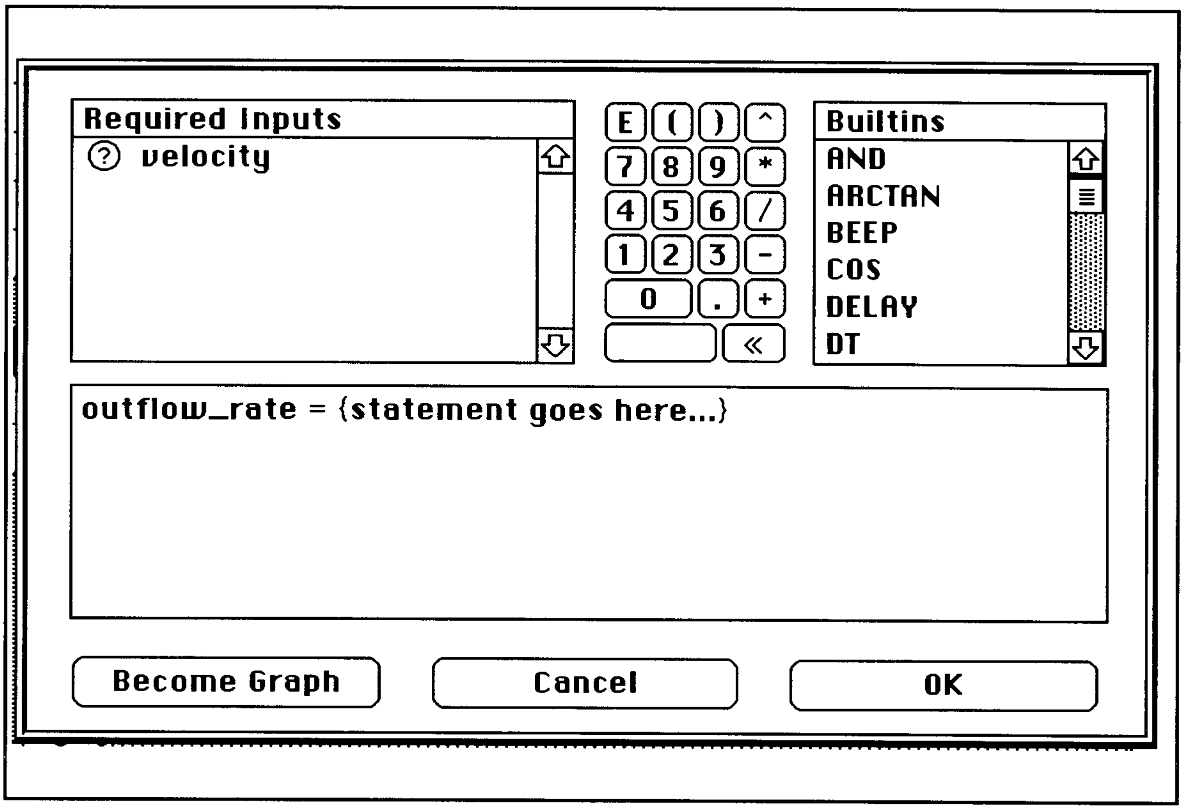


Figure 2.9: Defining the mathematical computation of a variable in STELLA

After specifying the mathematical relations among the elements of the model the system can simulate it by solving the finite difference equations using one of three possible numerical methods (which can also be chosen by the user): Euler, 2nd-order Runge-Kutta and 4th-order Runge-Kutta .

2.5.3 Discussion of STELLA System

The most beneficial aspect of STELLA is its exploitation of a graphic interface to represent and animate the structural diagram of a given problem. With such a system it is possible for the learner to visualise changes to the whole system over time, permitting him/her to have a better understanding of phenomena.

Another positive aspect of STELLA is the different outputs it can present for a certain simulation. Discussing and linking the ideas embedded in these different outputs can help the process of creating meaning from different representations (Teodoro, 1994).

However, in respect to the interface, three main problems can be pointed out. First, the objects that represent the variables of the problem being modelled (stocks, flows and converters) do not present the initial quantities attached to them by the user until he/she asks for the system to run the model. This can be cumbersome for the learner because he/she can not have a qualitative idea about the initial state of the model before starting a simulation.

The second problem is that although the system permits the user to set individual scales for the variables, it does not explicitly show these scales when the software is running. Again this can confuse the user specially in situations where he/she has many objects on the screen.

The third problem is about RUNNING a model. Although the user can set the number of iterations for a certain simulation, the system does not provide a way to control how fast it is going to run a simulation. So, if the user creates a model with just three or four variables it will be very difficult to visually keep track of the changes of the variables while the model is running. This becomes specially difficult using a fast machine.

The next three problems are more fundamental and have to do with how the system was conceptualised and with how the visual metaphor is used to represent models.

The system's conceptual model is based on a metaphor of values flowing in and/or out of stocks. To represent these two kinds of flow, the system uses an icon called *flow* which resembles an arrow indicating the direction of flow. So a flow starting on a stock and going to the clouds gives the expectation that the stock is being de-accumulated (an out-flow). A flow finishing on a stock (an in-flow) gives the expectation of a stock accumulating. However if the user gives negative values for the flows, they will behave in the contrary way: an in-flow with a negative value will make the stock empty. In the same way, an out-flow with a negative value will make the stock fill up. So what we have here is a compromise between the mathematics of the system and the way it is presented by the interface. If we recall that STELLA was developed for students being initiated into modelling, then the visual representation of a model must have much more importance than the mathematics that goes behind it. As a result, at least for beginners, the model presented in Figure 2.10 can have a certain inconsistency between what is shown on the screen and the results of its simulation (with *Mortality* being negative, *Population* will increase instead of decreasing).

Another inconsistency between the mathematics and the visual metaphor of the system is that although it is possible to have negative values in the underlying model (at the mathematical level), the minimum visual value for a stock on the screen is zero ('empty').

It is also true that the visual metaphor used by STELLA is not appropriate for certain kind of problems. For instance *velocity* and *displacement* are two physical entities not at all obviously appropriately represented by a tank, though in STELLA they need to be. Although they can increase and decrease, people do not normally associate them with the idea of accumulating or deaccumulating.

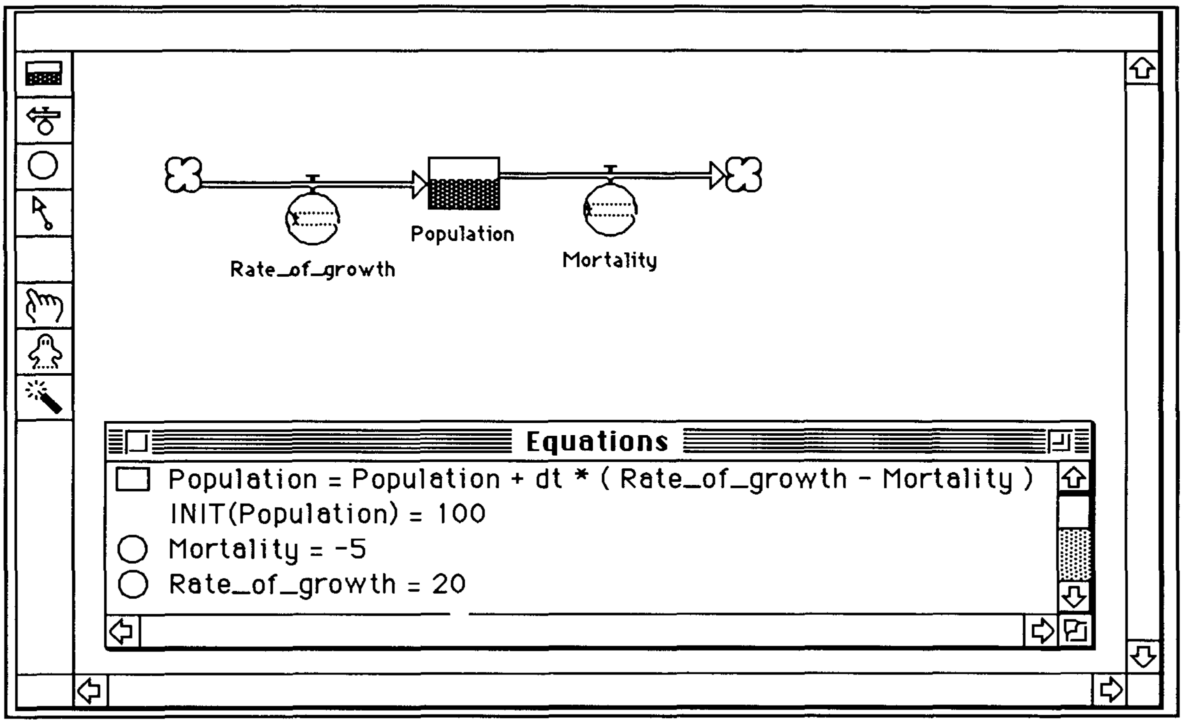


Figure 2.10: The underlying model is not consistent with the visual presentation

Another negative aspect of STELLA is the fact that it is out of the reach of younger students due to the mathematics required to create models. Although phenomena involving simple algebraic functions and first order equations can be picked up by 12-15 years old students, it becomes difficult to use the system if the mathematical model has to involve higher order equations and/or quadratic, logarithmic and sinusoidal (etc.) functions. In this case the only possibility open to the students is to play with simulations already created by their teachers.

2.6 IQON SYSTEM

The ESRC "Tools for Exploratory Learning" project (Bliss & Ogborn, 1989) proposed to develop a semi-quantitative modelling tool - IQON (Interactive Quantities Omitting Numbers) - as part of a Research Programme aiming to investigate the uses of modelling systems by students.

The main idea of IQON is to provide a modelling environment where the user can think in a semi-quantitative way (see Sections 1.5.4 and 3.4 for a discussion of semi-quantitative thinking and its relation to work in Qualitative Reasoning and Artificial Intelligence).

Within the IQON environment it is not necessary to specify the values of QUANTITIES AND ANALYTICAL relations between variables. The user has just to recognise changes and direction of changes between variables by relating them in a diagrammatic way. After having constructed the model, the user can ask for a simulation. At that time, the system "assumes the control of the model", being responsible for automatically attaching a set of dynamic mathematical equations (hidden from the user) that will govern the simulation. Following this, it makes the model iterate over time, giving visual feedback on the screen by changing the levels of the appropriate variables existing there.

The present version of IQON is a prototype developed in SMALLTALK for Macintosh computers using system 6.

The next two subsections present the IQON interface together with its use and offer a critique of the software.

2.6.1 IQON: General Description

IQON uses a direct manipulation interface to implement and interpret causal-loop diagrams in a consistent way (compare the model presented in Figure 2.11 with the one presented in Figure 2.12). When the system is loaded it presents the user with a window where models can be constructed or loaded (see Figure 2.11). At the top of this window there is a control panel with 13 icons that are for: (i) creating and manipulating the building blocks on the screen (9 first icons); (ii) running a model (3 icons) ; and file manipulation (1 icon).

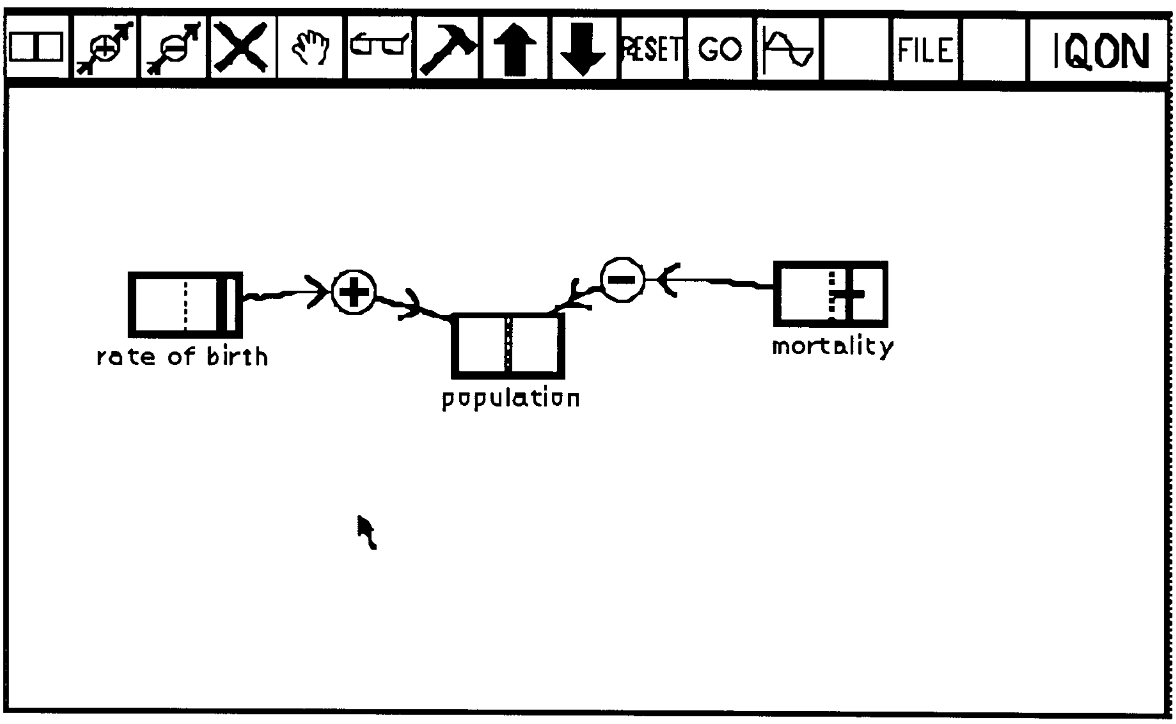


Figure 2.11: IQON screen with a simple model about Population

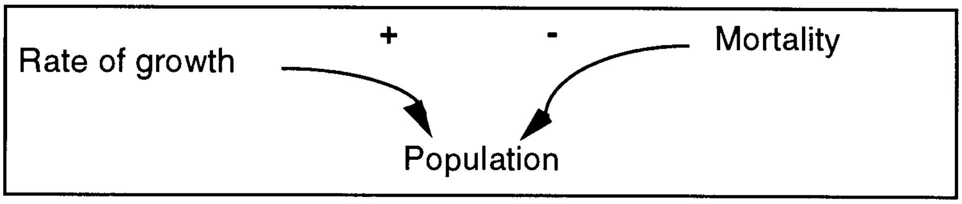
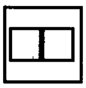


Figure 2.12: A causal diagram corresponding to the model shown in Figure 2.11

The system has two kinds of primitives (or building blocks): a *continuous-valued* variable (also called "box") that can assume semi-quantitative values compared to a *normal* level; and *positively affects* and *negatively affects* links to represent incremental changes between variables.

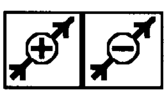
To create an object or to perform some operations on the objects on the screen, the user has to "press" an appropriate button on the control panel. Once an icon is selected, the system enters in a mode of operation associated with that icon. To change the mode, the user has to click on another icon (which will select another mode) or in the case where the system is in a 'go'/'reset'/'graph' mode, the user can click anywhere on the working area to leave that mode.

Presented below is a description of the main buttons and the operations associated with them (see Figure 2.11).



The 'box' button permits the user to create a continuous variable on the screen. All variables have to have a name which has to be given before the variable is shown on the screen.

A continuous variable contains a level bar that can be moved -either by the user or by the system when running the model - to the left or to the right. When it is in the middle of the box it is said to be in a resting state (indicated by a dotted line) and does not exert any influence on the dependent variables. The greater the deviation of the level bar to the right the bigger the value represented by the box. It is the same the other way around: the greater the deviation of level bar to the left the smaller the value represented by the box. The response of a level is non linear, limited by the ends of a box, which represent values as far above or below the 'normal' level as one cares to imagine.



Links are used to represent causal relations between variables and can pass positive or negative influences between variables. The way they affect a certain variable (by incrementing or decrementing its level bar) is determined by their sign: In the example shown in Figure 2.11 *rate of birth* positively affects *population* and because *rate of birth* is high then it will push up *population* . However if it was the case that *rate of birth* was low then population would be pushed down. A negative link works the other way: In the same example *mortality* negatively affects population and because it is high then it will push down population. If it was the case of *mortality* being low then it would push up *population.*

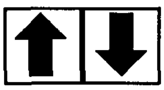
Links can also have different relative strengths implying a strong effect between a certain variable onto another. The system represents this idea by using bold '+' and '-' signs.


The 'cross' icon is used to delete variables and/or links. If a dependent variable is destroyed all of its incoming links are also automatically erased by the system.


The 'hand' button serves to reposition boxes and links on the screen. After selecting it on the control panel, the user has to click and drag the object he/she wants to move.


The 'spectacles' icon is used to give access to the internal parameters of a variable (its name and comments about it) also permitting the user to change these parameters.


The 'hammer' button is used to fix/liberate the level bar of a variable. When the level bar of a variable is fixed, its value does not change during a simulation. If the variable is a dependent one, all its links are redrawn using a shaded pattern to indicate that they will not work (as if they were "sleeping') during the simulation.


The 'up' and 'down' arrows permit the user to increase and decrease the level bar of the variables shown on the screen. After selecting the arrow on the control panel, the user has to select the variable he/she wants to change by clicking on it. Each new click on a variable will make its level bar move up or down. These arrows can also be used to change the sign of links. The up arrow will make negative links become positive and the down arrow will make positive links become negative.

By pressing the 'GO' button the model is made to run. When it enters this state the system automatically generates a quantitative model (a set of algebraic or differential equations always hidden from the user) based on the diagram on the screen. During the running process, each variable averages its input (In case it has more than one input) taking into account the relevant signs of the ingoing links to calculate its value for the next iteration. To stop running a model, the user can click anywhere on the screen.

The 'reset' button resets the system clock (which does not appear on the screen) and makes all variables go back to normal position (internal value = 0).


The graph button permits the user to visualise the variation of a certain variable (chosen by clicking on the desired variable) against time. The graph is not constructed in real time and only the result of simulation for one variable at each time is available (see Figure 2.13).

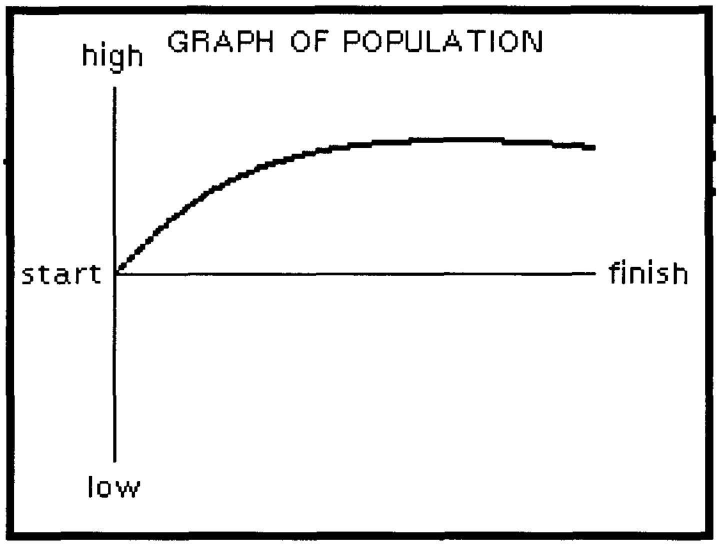


Figure 2.13: Graph of the variable Population X Time for the model shown in Figure 2.11

2.6.2 <u>Mathematics of IQON System</u>

In IQON the mathematics that controls the behaviour of the variables during a simulation is always hidden from the user. He/she is always thinking about both the model structure and its behaviour purely in terms of qualitative and semi-quantitative ideas. According to the developers of the system (Miller, Ogborn, Turner, Briggs, & Brough, 1990) its mathematics was partially inspired by looking at the mathematics used in developing connectionist nets (Rumelhart & McLelland, 1987) The properties of those systems used in IQON are: (i) they have a resting level; (ii) they respond over a limited range of values so that they can be interpreted as big or small values; (iii) they respond to inputs in a simple and uniform way by taking a weighted sum of them[3] (Miller & Ogborn, 1993). The Figure 2.14 below shows the correspondence between an IQON model and the mathematics that control its behaviour.

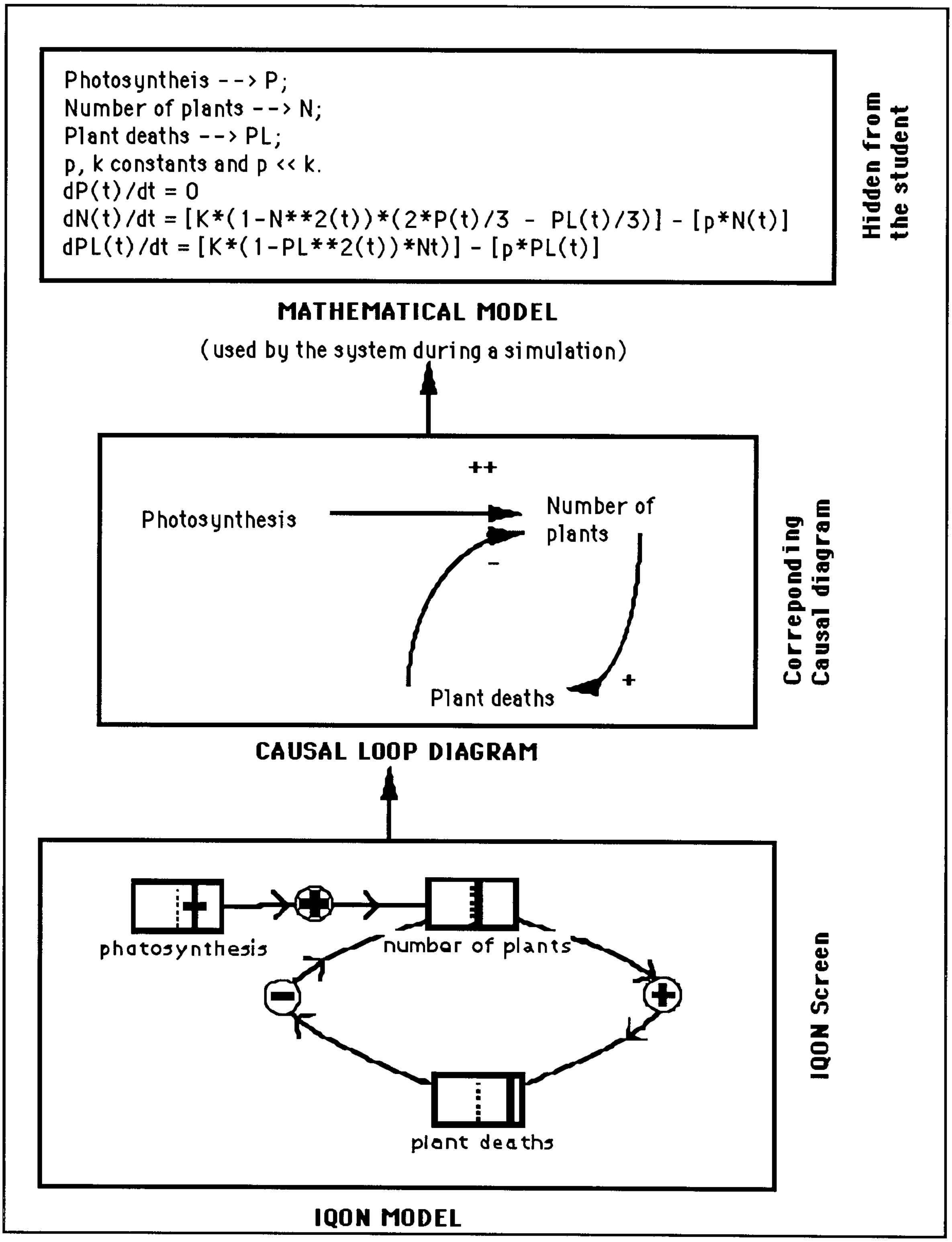


Figure 2.14: Correspondence between an IQON model, Causal loop diagram and the underlying mathematical model

3 A complete description of IQON's mathematics is presented in Miller & Ogborn (1993).

2.6.3 Discussion of IQON System

It is well known that when someone is trying to understand a situation, he/she normally has too little knowledge to form an exact quantitative model (Dillon, 1994). So, starting work with quantitative modelling to think about the dynamic behaviour of systems does not seem to be the appropriate way because in such situations the user is obliged to focus on the functional behaviour of the variables before having a systemic view of the problem. IQON came exactly to fill this gap by opening up the possibility for the students to create models without having to be mathematically precise about the relations between their constituent elements (Ogborn, 1990).

Another important aspect of such a system is that it permits younger students (12-15 years old) to be initiated into modelling activity. The reason is the same as that stated above: there is no need to know formal mathematics in order to use the system (Bliss, Ogborn, Boohan, Briggs, Brosnan, Brough, et al., 1992c).

In order to make this section more intelligible I will discuss below the aspects related to the interface of the software and its modelling possibilities in two different subsections.

2.6.3.1 IQON: Discussing its Interface

A very positive aspect of the system is its simplicity for qualitatively representing a certain phenomenon. The metaphor of *boxes* and *arrows* is very powerful to represent the idea of values and relationships. In a certain way we are accustomed to seeing this kind of diagrammatic representation in different sorts of printed material such as books, newspapers and folders. It can also be easily mapped onto the idea of causal loop diagrams with *boxes* representing system variables and *arrows* the relationships between them.

Compared to STELLA's metaphor of *stocks* and *pipes* the one used by IQON is simpler and yet more general (it can be seen as a superset of the metaphor used by STELLA). A good example is to compare the representation of physical entities such as *velocity* and *displacement.* With IQON the level bar just moving across the box can give the user the idea of something increasing or decreasing instead of filling or emptying a container. When it is the case of modelling a variable that accumulates and/or de-accumulates something the visual effect of the level bar combined with the variable-box can also be easily understood as a container being filled or emptied.

Another important aspect of IQON's metaphor is that it permits the visual representation of negative values as something 'being below the normal level'.

STELLA again, as mentioned above, does not visually provide the user the idea of negative numbers.

On the other hand, a negative aspect of the interface is that it requires excessive mouse operations to perform certain interface tasks. For example, when the user wants to move the level bar up or down, instead of directly going to the level, clicking on it and dragging it to the right or to the left, the user has first to select the appropriate button on the control panel ('up' or 'down' arrow) and then click on the desired variable as many times as the number of increments/decrements he wants. Worse still, if in the middle of this operation the user wants to move the level in the contrary direction, he/she has to click again on the other 'arrow' button on the control panel and repeat again the process of clicking on the variable.

Although the system provides the user with a plotted graphic as another way to represent the output of a variable, it is nevertheless very limited. The graph only shows the variation of the variable against time for a certain pre-determined period of time which is not known by the user. The graph is not constructed in real time and can only be presented after a simulation has been tried.

Another negative aspect is that the graph of any variable presents the internal value of the variable, using the hidden damping function (employed by the system as a relaxation function to make the system stable) as part of the calculation of the variable. So if the user defines that *population* increases with *rate of birth* and he/she only sees the graph of *population* versus *time* (Compare the model shown in Figure 2.11 with the graph output shown in Figure 2.13), he/she will have the wrong impression that *population* stops increasing after a certain period of time.

2.6.3.2 IQON: Discussing its Modelling Possibilities

Although IQON permits the user to model a wide range of problems related to the secondary school and undergraduate curriculum (Kurtz dos Santos, 1992), there are particular examples of these problems that are not well handled by the system:

• Conditional events - The occurrence of some events is sometimes conditioned by other events. Mathematical formalism permits to represent these situations by using constraints within a set of equations. A typical example is the fact that certain ecosystems like a lake can adapt themselves to the release of pollutants up to a certain level. If the concentration of the pollutant in the water becomes greater than a certain value, the animals and plants that are dependent on that environment begin to die or to have severe problems[4].

• Direct relationships - Not all relations in a dynamic system are the rate of change type. Actually there exist some relations that are functional and can be represented by algebraic equations. A typical case is the relation between the physical entities *force* and *acceleration.* As it is well known in dynamics they are associated by the equation $F = m * a$.

• Combination of inputs - There are many problems in nature that are not appropriately modelled by using the 'average' combination provided in IQON. A typical example is rust. It is a known fact that in order for rust to occur (for instance in a car) two components are necessary: water and oxygen. If in a certain environment only one of these components exist then there will be no rust. The present input combination used by IQON is not appropriate to model such situation because it averages the inputs of a certain variable no matter what their values. Another situation would be when the user wants to add the causal factors to calculate the value of a dependent variable. In the example given above about lake pollution, it is evident that if more than one pollutant is being released in the lake the elapsed time until the lake becomes "polluted" would be proportional to the sum of the pollutants and not to their average[5] .

• 'Positive/negative only' variables - Variables that have values that can be above or below a certain level are not the rule for all problems in nature. The example above about the lake pollution serves as a good example here as well. There is no sense to the idea that a lake is polluted 'below the normal'. We know that either a lake is not polluted or it is polluted at a certain level. In this example the normal level should be interpreted as the smallest possible value for the pollution of the lake. This is the same case for rust (if someone is analysing the problem from a chemical point of view). Either it exists or not. There is no "real world interpretation" for rust being 'below normal' (or less than zero). The system should provide a way for the user to say so.

2.7 SCIENCEWORKS MODELER SYSTEM

Some time after the main design decisions about LinkIt had been made, I became aware of a different approach to semi-quantitative modelling at the University of Michigan (USA). A software first called ScienceWorks Modeler and

---

[4] The idea of simulating conditional events is the focus of Discrete Event simulations. Within this category the modelling primitives include the notions of events and delays instead of variable-type primitives.

[5] It is also true that the time is also related to the type of pollutants. But even if someone makes an approximation considering that the pollutants have the same "power of pollution", IQON can present very unexpected results for certain combinations of inputs.

later Model-It was developed as part of a project integrating computing technologies into the daily life activities of students. (Jackson, Hu, & Soloway, 1994; Jackson, Stratford, Krajcik, & Soloway, 1995). The software was developed in Smalltalk for Macintoshes only. The discussion presented below is based on a demo version of ScienceWorks Modeler (Version 1.0a), through personal electronic communication with one of the authors and the articles mentioned above. The Figures shown here were taken from the demo version. Whenever a major difference between the two versions exists, a comment about it will be added.

2.7.1 ScienceWorks Modeler: General Description

ScienceWorks Modeler provides a direct manipulation interface to build a wide range of process flow models. To assist the students in this process the system provides a set of high level objects that take the form of digitized photographs and graphics. To construct a model students have to create or select from a set of objects the ones related to the problem to be modelled and define measurable quantities or values (factors) associated with these objects (see Figure 2.15). To define the factors corresponding to each object and the relationship between them, the user has to call the object editor (see Figure 2.16) and put the appropriate factors into the small boxes called "... Is affected by..." and "...and affects...".

Following this, students have to define the type of relationship between the factors. This can be done by evoking the "Relationship Maker" window (see Figure 2.17). The definition of a relationship is made through a qualitative verbal representation by selecting descriptors in a sentence. The system basically permits two types of relationships: absolute and rate of change (absolute relations can be linear, exponential or a "bell curve") and permits two directions of change: "increases" and "decreases".

Before simulating the model the user has to define the interval of variation and the initial value of each factor involved in the model. This can be done by the "Factor Factory" window (see Figure 2.18).

When the model is running the values of the factors can be presented on the main window (under user discretion) via small windows containing columns that are filled or emptied. Figure 2.17 shows the values of two factors *MidgeFly:count* and *Stream:oxygen*. after 21 iterations.

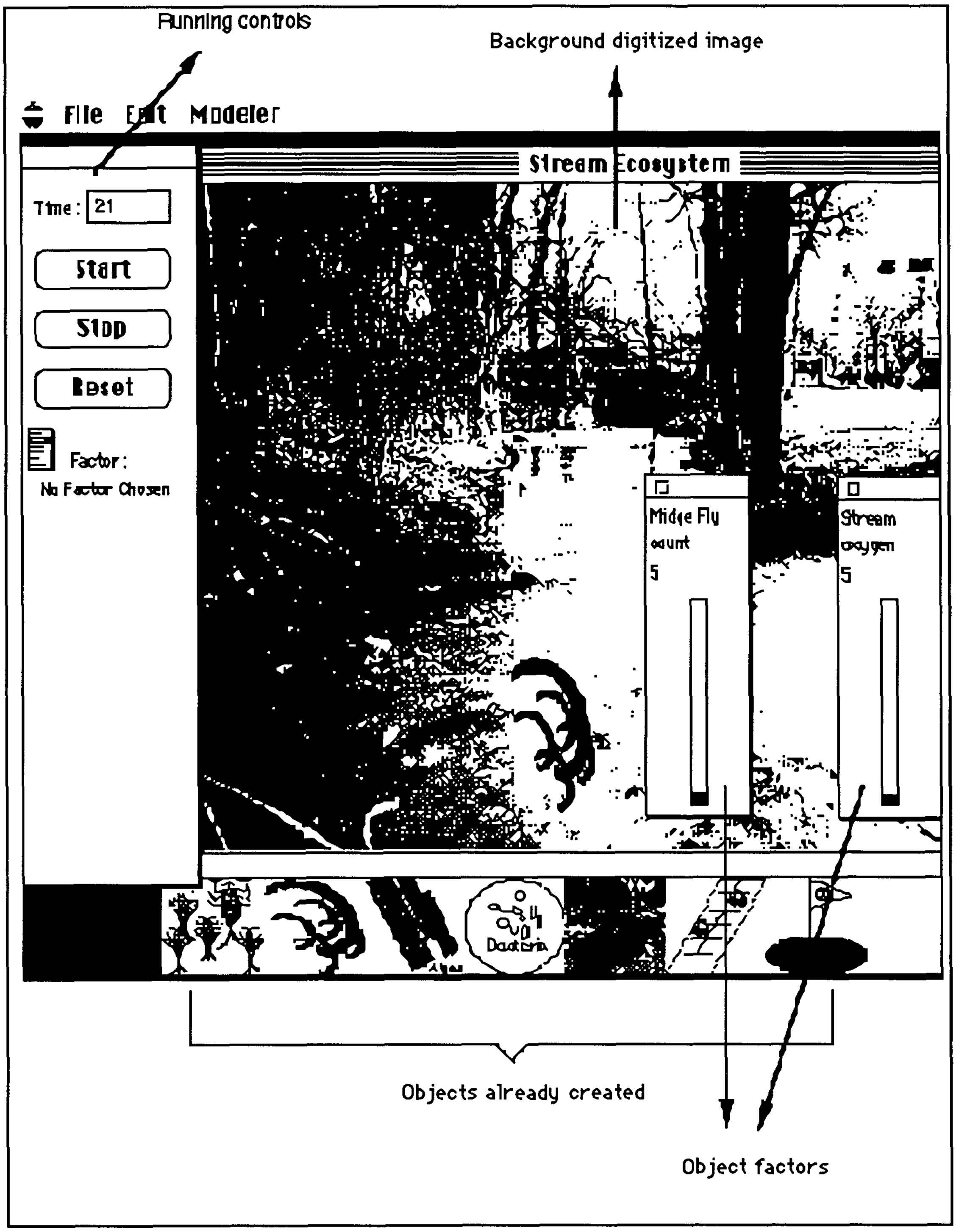


Figure 2.15: Main window of ScienceWorks Modeler showing a digitized image of a river and two factors (Midge Fly:count and Stream:oxygen) related to this problem

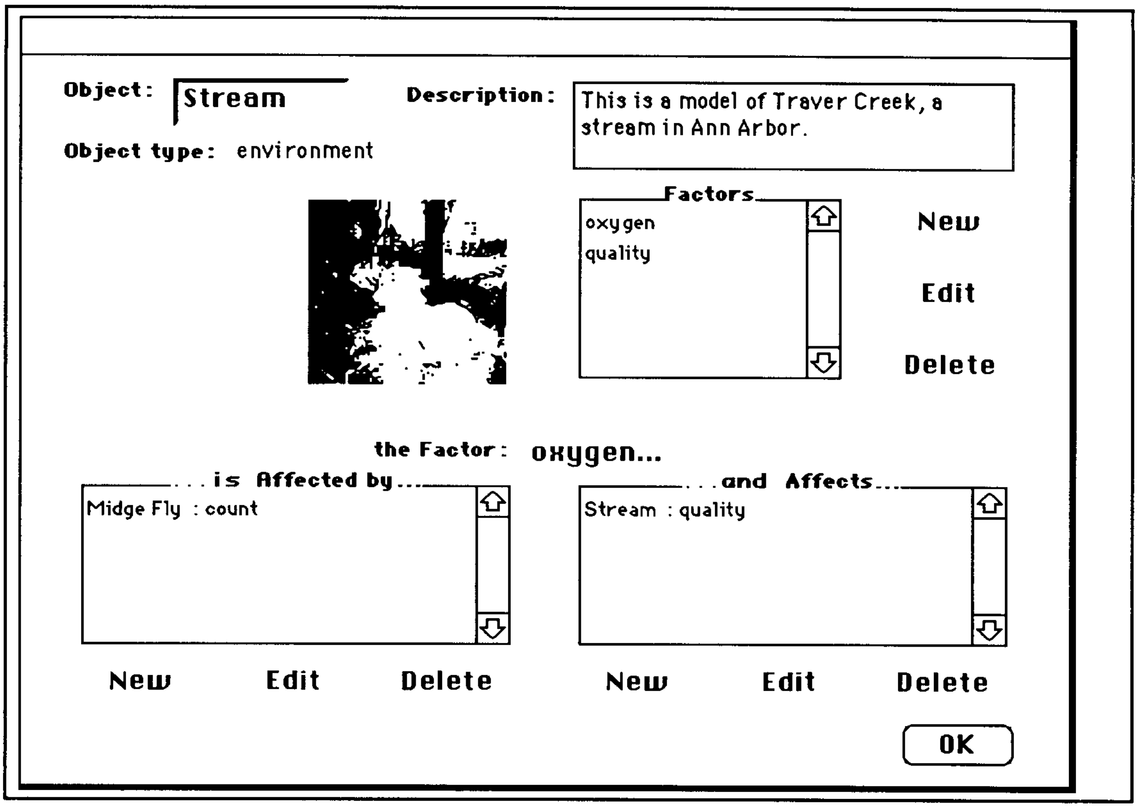


Figure 2.16: Object editor window showing the factors of the object *stream*. According to this window the factor *oxygen* is affected by the factor *MidgeFly:count* and affects the factor *Stream:quality.*

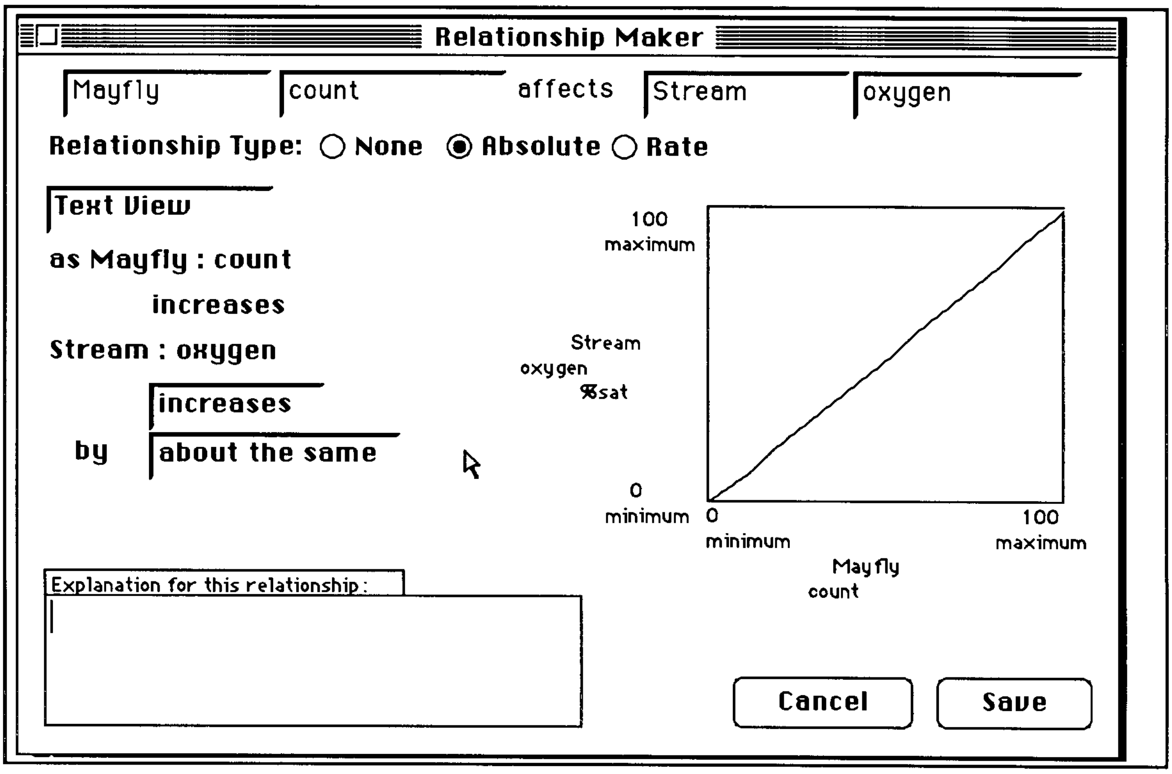


Figure 2.17: Relationship Maker window. Specifies the type of relationship between factors. Here the factor *Mayfly:count* affects the factor *Stream:oxigen* by increasing it in a linear way[6]

---

[6] In the new version of the system (Model-It) the name *absolute* relationship was changed to *immediate* relationship. It also permits the user to define relations between factors by filling a table of numerical values.

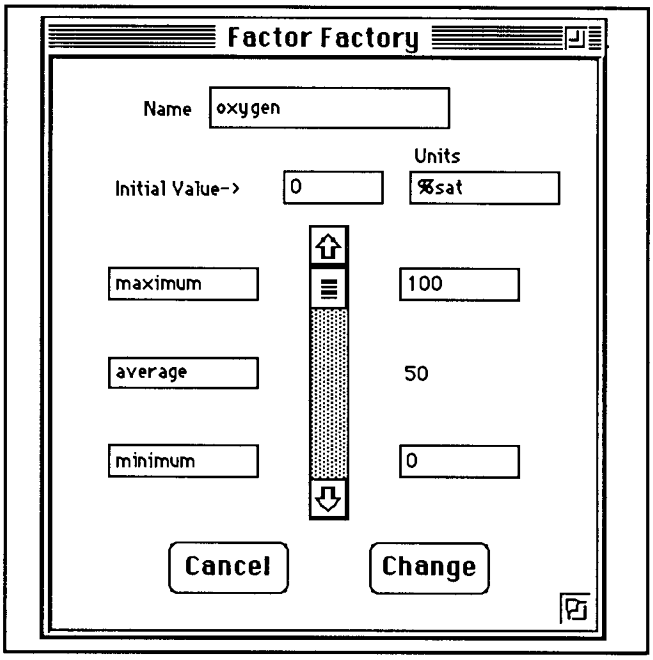


Figure 2.18: Factor Factory window. Used to define the range of variation and initial values of a certain factor

2.7.2 Discussion of ScienceWorks Modeler System

The most important aspect of this system - in comparison to STELLA and IQON - is the possibility of mixing a digitized image with the important factors about a certain problem. This can assist the learner to make the transition from what he/she already knows of the world over to computerized model representations.

The possibility of defining relationships based on predefined plotted graphs also gives the user a chance to perceive different representations of a certain set of data. However this way of relating variables can be useless for young students due to their lack of knowledge of cartesian representations.

Contrary to STELLA and IQON the first version of the system does not permit the users to have a structural and relational view of the model. The new version discussed in Jackson et al. (1995) incorporates this facility. However it is done in a limited way, only showing a static map of the relations.

The way the system presents the value of the factors during a simulation is not easily interpreted, specially when the user defines factors that can assume negative values. In this case the system does not provide a clear indication of where the boundaries of the negative and positive intervals are.

Another negative aspect of the software is a consequence of the metaphor employed to represent the dynamics of the variables involved. The system uses the idea of "filling/emptying a colunm" (in a similar way to STELLA) which is not appropriate to model certain kinds of problems (see Section 2.5.3 about the problems with STELLA).

2.8 FINAL REMARKS

Different researches either employing (in school and undergraduate courses) or analysing the use of computational systems within a systems dynamic approach perspective have pointed to positive aspects of these systems as educational tools. Some of these aspects are:

- To make explicit one's assumptions about dynamic systems and bring him/her to a better understanding of the system's behaviour
- To help students to engage in the process of modelling
- To enhance the acquisition of declarative and procedural knowledge
- To stimulate cross-course exploration of phenomena.

Three of these systems have been described here: STELLA (Lindfield, 1992; Mandinach, 1989; Riley, 1990; Santos & Ogborn, 1994; Schecker, 1993; Steed, 1992; Whitfield, 1988) for being the first computer dynamic modelling system with an educational purpose putting together the ideas of direct manipulation interfaces and systems thinking; IQON (Bliss & Ogborn, 1989; Kurtz dos Santos & Ogborn, 1994) for combining these resources with ideas of qualitative reasoning and Artificial Intelligence to present an environment where no formal mathematics is needed to construct models, giving younger learners the opportunity to be initiated into the ideas of modelling; and ScienceWorks Modeler for taking a different approach (from IQON) to represent qualitative reasoning.

The work presented in this thesis starts from the basic ideas that support STELLA and IQON systems[7] to construct and test a new computer modelling tool that goes beyond the limits presented by these systems.

---

[7] As said in the beginning of Section 2.7 I became aware of ScienceWorks Modeler only after the main design decisions about LinkIt had been made.

Chapter 3:

RATIONALE FOR THE RESEARCH

3.1 INTRODUCTION

The rationale for this research is constructed around four main premises:

1) People have mental representations about situations and/or problems of their daily lives

2) Learning is an active process of exploration/improvement of these mental representations

3) Semi-quantitative thoughts are crucial in generating an acceptable account of what is going on in a certain phenomenon

4) Computer modelling provides a way for mental representations to be externalised and tested.

In this chapter each of the above premises is discussed in a separate section. The chapter then continues with some general implications for the design of the software and concludes with the main ideas for the research and its general research questions.

3.2 MENTAL MODELS

The process of thinking has been the concern of psychologists since the beginning of this century. Some are more interested in what happens when people are thinking, while others try to explain how cognitive abilities develop. Despite the different interests in these two strands of research, there are some similarities in their work. For example, the similarity between the account given by Piaget about the concrete operational stage of reasoning in children and Johnson-Laird's mental models theory. Both see the process of thinking heavily based on the internal representation of objects and events.

Johnson-Laird (1983) in his book *Mental Models* provides an explanation of mental models:

> *If you know what causes a phenomenon, what results from it, how to influence, control, initiate, or prevent it, how it is related to other states of affairs or how it resembles them, how to predict its onset and course, what is its internal or underlying 'structure' is, then to some extent you understand it. The psychological core of understanding, I shall assume, consists in your having a 'working model' of the phenomenon in your mind* (p. 2)

Mental models are constructed by someone while he/she is interacting with the world. They evolve with time and have both explicative and predictive power. According to Craik (Craik, 1943) - the creator of the concept of mental models - the construction of mental models is at the heart of the process of thinking, acting as a mediator between perception and action, providing the means to interpret, to remember, to communicate information and to control performance (Bliss, Mellar, & Ogborn, 1992a; Wild, 1996).

Johnson-Laird (1983) also gives an account of what constitutes mental models, talking about the relation between what is intended to be represented and what is there in the mind:

> *... each entity is represented by a corresponding token in a mental model; the properties of entities are represented by the properties of their tokens; and, relations among entities are represented by relations among their tokens* (p. 16).

The evolution of a mental model is one consequence of the interaction between the person (who holds the mental model) and the world (including the person's inner thoughts). In this way these models need not be complete or technically and scientifically correct (Norman, 1983). As mentioned above, they exist to explain and to help predict situations. When they do not serve this function they need to be reformulated in order to respond more appropriately to the behaviour of the environment.

3.3 MENTAL MODELS AND LEARNING

If one accepts this theory about how knowledge is represented and is concerned about learning and education, a possible question to be posed is: what are the implications of this theory for learning ? According to this theory, learning can be seen as a process of extending and broadening someone's existing mental models in order that "(the person) *reacts in a much fuller, safer and more competent manner to the emergencies that face him/her*" (Hebenstreit, 1987). Within this framework the process of learning is one of active construction of knowledge (by the subject) - as opposed to a merely passive transmission of knowledge (in the strict sense) - in which learners have the possibility of acquiring a profound insight into the ideas they are exploring:

> *Contemporary perspectives on cognition reject the view that learning is a one way process whereby the learner receives and organises stimuli from an external world. Instead learning is viewed as an active process in which the learner brings prior sets of ideas, schemes or mental representations to any interaction with the environment. These enable predictions and interpretations about features in the environment to be made, expectations generated and assessed and as a result the mental representation may be changed.*
>
> (Millar & Driver, 1987, p. 46-47)

In order for this construction to be effective it has to depart from what subjects already know. The new concepts to be assimilated into a learner's framework of knowledge are constructed through the establishment of links with other concepts (deCorte, 1990; Piaget, 1954; Resnick, 1983 cited in Hebenstreit, 1987). Results from research on situated cognition have demonstrated that learning is optimised when instruction is anchored in real world problem solving environments (Brown, Collins, & Duguid, 1989 cited in Clancey, 1992; Shute & Gawlick-Grendell, 1994). Nevertheless, for a long time teachers have been neglecting this idea, trying to teach students new scientific thoughts without paying attention to the theories they already possess (for some research about commonsense reasoning see Driver, (1983); Gentner & Stevens, (1983)). This process also has overemphasised the use of mathematical formulas and procedures to solve problems to the detriment of qualitative (actually semi-quantitative) ways of describing and relating physical phenomena and mathematical ideas.

Dillon (1994) in his overview of qualitative reasoning research, when discussing the differences between quantitative and qualitative reasoning, has pointed out that the "*formal analytic approach gives little insight into how the* *behaviour represented by the solutions actually comes about*". However, qualitative analysis "*captures important information about how and why changes occur*". These ideas support what some research in science teaching suggests needs to be changed in science and mathematics education (DiSessa, 1987; Ogborn, 1990): "*Change from learning symbols and algorithms without meaning to learning meaning before symbols and algorithms*" (Teodoro, 1992a, p. 14).

3.4 THE IMPORTANCE OF SEMI-QUANTITATIVE THINKING

Semi-quantitative thinking is very important for giving an account of daily-life phenomena. Even when people are presented with quantitative results, they have a tendency to reinterpret them in terms of semi-quantitative ideas. If, for instance, someone is shown a quantitative model about chemical pollutants released in the environment and their relation to the greenhouse effect, what is likely to be in the back of her/his mind is something like "An increase in the pollutants makes the greenhouse effect worse". So semi-quantitative thoughts seem to be a natural way of reasoning. Another important aspect is that even when someone is considered an expert in a certain subject he/she tends to communicate his/her ideas about it using semi-quantitative reasoning. The use of quantitative descriptions is restricted to situations where a high level of conciseness and a formal analytic approach are desired (Dillon, 1994).

The idea of semi-quantitative thinking has gained importance in the last two decades and has been the subject of study by researchers in the Artificial Intelligence field in which context it is known as 'qualitative reasoning' (Q.R.) (de Kleer & Brown, 1981; de Kleer & Brown, 1983; Forbus, 1984; Kuipers, 1994) [1]. The aim of these research programmes is to identify what constitutes the "core knowledge" - also known as "commonsense theories" - that underlies the physical intuition of ordinary people. This knowledge, as Hobbs points out in his edited book *Formal Theories of the Commonsense World* (Hobbs & Moore, 1985), is quite different from the way science perceives it. He stresses the importance of the Q. R. field, suggesting the construction of an intelligent agent (a robot which eventually would be controlled by a computer program), and asking himself what kind of knowledge this robot should possess in order to be able to go to the cafeteria to buy a portion of salad and a sandwich.

The different approaches to the A. I. field reflect the richness of the area and the many different kinds of "commonsense knowledge" we possess. To take an example, a short term goal of de Kleer & Brown's work is to construct computer programs capable of generating qualitative descriptions of a physical

---

[1] An approach to this field from a more educational perspective is known as 'commonsense reasoning' or 'alternative conceptions' (Driver, 1983; Driver, 1989; Gutierrez & Ogborn, 1992).

device knowing the behaviour of its parts (de Kleer & Brown, 1981; de Kleer & Brown, 1983). In doing so they are pursuing the construction of what they call a *qualitative causal physics.* This "new" physics provides an alternate and simpler way of arriving at the same conceptions and distinctions addressed by traditional physics, thus possibly providing a much simpler pedagogical basis for teaching/learning about physical mechanisms (de Kleer & Brown, 1985).

In a certain way researchers in the Q.R. area are looking for what mental representations people use to make their way in the world. Although the research about mental models in the field of psychology has a lot to do with this idea, there are some important aspects that differentiate these two strands of research. The ultimate goal of mental models research in cognitive science is to develop educational techniques to help novices become experts in a certain area of knowledge. But for qualitative reasoning researchers the aim is to construct machines (or computer programs) that behave as novices. Another important difference is that the cognitivists are not required to think about mental models in a very formal way (until a certain level). However, constrained by the fact that computer programs need to be constructed to simulate certain behaviours, people in the Q.R. field need to define these models using a certain precise logical notation.

3.5 EXTERNALISATION OF THINKING

It is commonsense that externalising ideas helps us to think about them. The most common way of doing it is through speech and action[2]. However, these ways of thinking are ephemeral and do not permit the storage of this information for later use, nor a deeper reflection on what is being thought. Therefore, throughout their history, human beings have been developing external mechanisms that work as "registers of thinking" . Examples are writing, paintings, dance, music, etc.

The advent of computers provided another dimension for these mechanisms: they not only store information but can also process it, generating new information. In that way the computer can be regarded as a partner, helping us with the process of thinking. That may be why its use for educational purposes is almost as old as its implementation (Solomon, 1987).

However, for the scope of this work there is a particularly important feature of these machines that can help the process of thinking: they not only provide a way for someone to externalise his/her mental representations but they can also run these representations, giving the person continuous and direct feedback

[2] Vygotsky in his book *Thought and Language* (1962) gives great importance to speech saying that "*Thought is not merely expressed through words: it comes into existence through them*".

about his/her ideas in a certain domain. These permit an objective manipulation and observation of the models, opening the possibilities for experimental testing of one's own cognitive structures in terms of concrete models (Gorny, 1988; Webb & Hassell, 1988).

3.6 IMPLICATIONS FOR THE DESIGN OF THE SYSTEM

There are no definite rules about what has to constitute an educational computer system in general and a computer modelling system for learners in particular. The main issue that has to be considered when developing software for learners is that the user can be naive both in respect of the underlying concepts to be taught and in respect of the use of the program (Nicol, 1990). Therefore research in the area suggests that theories from cognitive science, pedagogy and computer science have to be taken into account when designing computer software for learners (deCorte, 1990; Gorny, 1988; Kozma, 1987).

The four sections above described the premises that form the theoretical basis for the design of the computer system studied here. They also point to some learning requirements that need to be considered in its development:

- Cognitive growth - Defined as the acquisition/improvement of content knowledge and analytic problem solving skills by the students. In order to promote cognitive growth, software needs to represent information in a familiar way and at the same time help to introduce more professional and symbolic representations (Jackson, et al., 1995) .
- Diversity of teaching-learning styles - Different approaches for teaching-learning can lead to the development of different knowledge. For instance Bliss et al (Bliss, et al., 1992c) suggest two modes of learning: exploratory and expressive. The exploratory mode permits students to investigate other people's ideas about a certain domain, which are quite often different from their own ideas about that domain. The expressive mode permits students to represent some of their own ideas about a certain domain, experimenting with and reflecting on them.
- High degree of control and interaction - By interacting with the structures of their models (and not only setting parameters in a simulation) and controlling their execution, students can explore what happens under different circumstances, try different approaches to thinking about a situation and develop a deeper awareness of their knowledge.

The theoretical basis discussed above and the pedagogical requirements that follow them imply a set of Human Computer Interaction (HCI) principles that are the key points for the development of the interface of the system.These principles can be summarised by the idea of giving a visual representation of the objects and actions of the world to be modelled. This idea is well known in HCI as a direct manipulation interface (or objects) and includes the concepts of WIMP (Windows, Icons, Mouse and Pull-down menus)-WYSIWYG (What You See Is What You Get) - MW (Multiple Windows). This style of interaction has two main benefits in comparison to a more command driven style of interface (Shneiderman, 1992):

• Gives the user a better sense of manipulating concrete objects - The actions on the objects on the screen are made directly on these objects and not through an indirect and more artificial way such as a command language.

• Reduces cognitive load - The users do not have to learn a new language to interact with the objects. They can start from their knowledge about interacting with objects in the real world to try things in the computer world.

3.7 GENERAL RESEARCH QUESTIONS

The central objective of this research is the design and implementation of a semi-quantitative computer modelling tool. This process could not be done without observing the intended end-users (secondary students) of the tool interacting with it. However, the studies carried out were not designed only to evaluate the software but also to shed some light on what could be expected from students when interacting with such a tool. In this way the general guiding research question of this work can be stated as follows:

> GIVEN A SEMI-QUANTITATIVE MODELLING TOOL (LinkIt), DOES IT PERMIT STUDENTS TO EXPRESS AND EXPLORE THEIR KNOWLEDGE IN A CERTAIN DOMAIN (E.G. SCIENCE) ?

It is also important to realise that a carefully designed educational tool is not enough to guarantee that students will learn what is intended to be taught. The preparation of lessons and the context where the tool will be employed are also of fundamental importance. Although how to use the tool and what should constitute a lesson aiming to introduce and explore the use of computer modelling systems in classroom activities is not the focus of discussion of this thesis, some of these issues are embedded in the way the tool was introduced and used with the students in the two studies carried out in Brazil.

From the point of view of designing a computer modelling system for educational purposes, the research question can be re-stated in the following way:

> HOW CAN SOMEONE'S SEMI-QUANTITATIVE IDEAS BE TRANSLATED INTO COMPUTATIONAL TERMS ?

This question is broken into more specific questions concerning the design of the software. These questions are presented in Chapter 5 together with the description of the Introductory study.

This work approaches the design and development of the software considering the learner as its central focus of attention (learner-centred design (Jackson, et al., 1995; Norman & Draper, 1986)). Within this framework and bearing in mind that the design process is seldom an orderly or linear process (Rubinstein & Herssh, 1984), it was decided to answer the questions above by first constructing a preliminary version of the software (described in Chapter 4) - called LinkIt I - and then conceiving an empirical study (Chapter 5) to observe students interacting with the system. The main aim of this Preliminary study was, therefore, to confront the system conceptual model with the user model through the system image (Booth, 1990; Norman, 1983; Norman & Draper, 1986), using the "think aloud" technique to expose students' mental processes (Gomoll, 1990; Sasse, et al, 1985).

The terms system conceptual model, user model and system image have different meanings for different authors. The definitions assumed here conforms with the definitions used by Norman in Norman & Draper (1986):

- System conceptual model - The model of the system conceptualized by the designer;
- User model - The mental model of the user about the system he/she is interacting with;
- System image (also known as target system) - Part of the software which is presented to the user.

The way the students employed LinkIt I to discuss some problems in science (also presented in Chapter 5) gave feedback for its redesign (Chapter 6), giving rise to a new version called LinkIt II. This new version was again tested by students (chapters 7 to 10). However this time, as stated earlier, not only was the design of the software considered, but also some educational issues are discussed.

From an educational point of view the general guiding research question above becomes an investigation of:

> HOW WELL DOES THE LINKIT REPRESENTATION WORK TO SUPPORT THINKING/LEARNING/EXPLORING IDEAS ?

Within this research this question is addressed in three forms (or along three dimensions). The first dimension is concerned with students' ideas about models. Here the aim of the analysis is to answer the following questions:

- What is the purpose of a model ?
- What is a good model for the students ?

The second dimension focuses on the models created by the students. The questions to be answered are:

- What is the level of complexity of the models created by the students ?
- What is the level of appropriateness of the models constructed ?
- Are there any patterns in the construction of the models ?
- How did improvement of the models occur ?
    - Nature of change
    - Process of change
- Are there any patterns in the construction of the models ?

The third dimension is about students creating models: what they said and did when interacting with the software to construct/explore models:

- Do the students engage in semi-quantitative thinking ?
- What ideas do the students initially have about a domain ?
    - Do their ideas develop as their work progresses ?
    - Do their ideas become better articulated ?
    - Do they think of new ideas (or change ideas) as their work progresses ?
- How do the student's ideas relate to the models ?
- Do the students see connections to other problems ?
- What relation do the students see between models and reality ?
- Are the students capable of comparing different models in relation to their underlying structures ?
- Do the students formulate/test hypotheses ?
    - How do they go about it ?
- Do the students analyse situations ?
- Are the students capable of detecting and diagnosing errors ?

These three dimensions are discussed in Chapter 10 although Chapter 9 - which mainly focuses on the learner-system interaction - also addresses some of the questions presented above.

PART II

DESIGN AND TESTING OF LINKIT I

The main ideas contained in LinkIt result from the studies done with IQON (Miller, Briggs, Brough, & Ogborn, 1992; Kurtz dos Santos, 1992). Nevertheless, when it was decided to develop a new version of that system, some new ideas that were not dealt with in the previous studies were considered and implemented. Basically these ideas consist of:

- The development of a new interface aiming to make the objects of the system easier to manipulate;
- The implementation of new features to make the system closer to users' ideas;
- Use of a computer environment which permits cross-platform versions;
- Use as little as possible of hardware resources.

After the development of the software it needed to be tested. So a Preliminary study was prepared mainly to check its viability, focusing on the interface and the components of the system.

This part of the thesis is about this first stage of development of LinkIt and is divided into two chapters. The first chapter (Chapter 4) presents the design and implementation of a first version of LinkIt. This is followed by Chapter 5, which presents a description of the Preliminary study and a synthesis of the analysis of the data collected.

Chapter 4:

LINKIT I: DESIGN AND IMPLEMENTATION

4.1 INTRODUCTION

In this chapter I shall describe the design and implementation of a first version of LinkIt (called LInkIt I) developed for this research. This version is for Macintoshes and it was implemented using the cT programming environment.

4.2 TENETS FOR THE CONSTRUCTION OF THE INTERFACE

There are two main features that are important to consider when producing software for learners: the pedagogical tenets (see Chapter 3) and the tools available (building blocks and interface) that can be used by the user to engage with a certain task. As a general rule it is said (Draper, et al., 1991; Shneiderman, 1992) that the more intuitive the interface - designed in such a way as to permit the learner to concentrate his/her effort on the task (low cognitive oveload) - the more friendly the software will be.

From this perspective the problem of creating educational software becomes a balance between the tools available for the user and the universe of possibilities of the software (i.e., the different categories of problems that can be modelled and their level of realism). Therefore, considering that our typical user is someone who is being introduced to thinking about/with models, and that in the future they might migrate to more quantitative systems, we opted for the development of an interface and a set of primitives that try to correspond intuitively to the way of talking about dynamic systems, but of sufficient generality to be applicable to a wide range of problems.

4.3 LINKIT'S INTERFACE: GENERAL DESCRIPTION

LinkIt is representative of a category of computer based modelling systems called *semi-quantitative* systems (see Sections 1.5.2 and 2.6). The system can be used to represent causal relationships between objects and/or events of a certain domain. The vocabulary used in the LinkIt environment tries to correspond well to intuitive or common-sense ways of talking about systems.

The construction of a model with LinkIt is made through the use of a direct-manipulation interface. This style of interface interaction permits the creation and manipulation of *concrete-abstract objects* (or screen objects). They are concrete in the sense that they can react to the user's actions through the use of the mouse or the keyboard. They are abstract in the sense that they just exist in the computer world (Teodoro, 1992a). In most cases these objects can be used by the modeller to represent some aspects of the world to be modelled such as the population of a city, the rate of growth of this population or the maximum number of people that could live in that city before it begins to have social problems (see Figure 4.1).

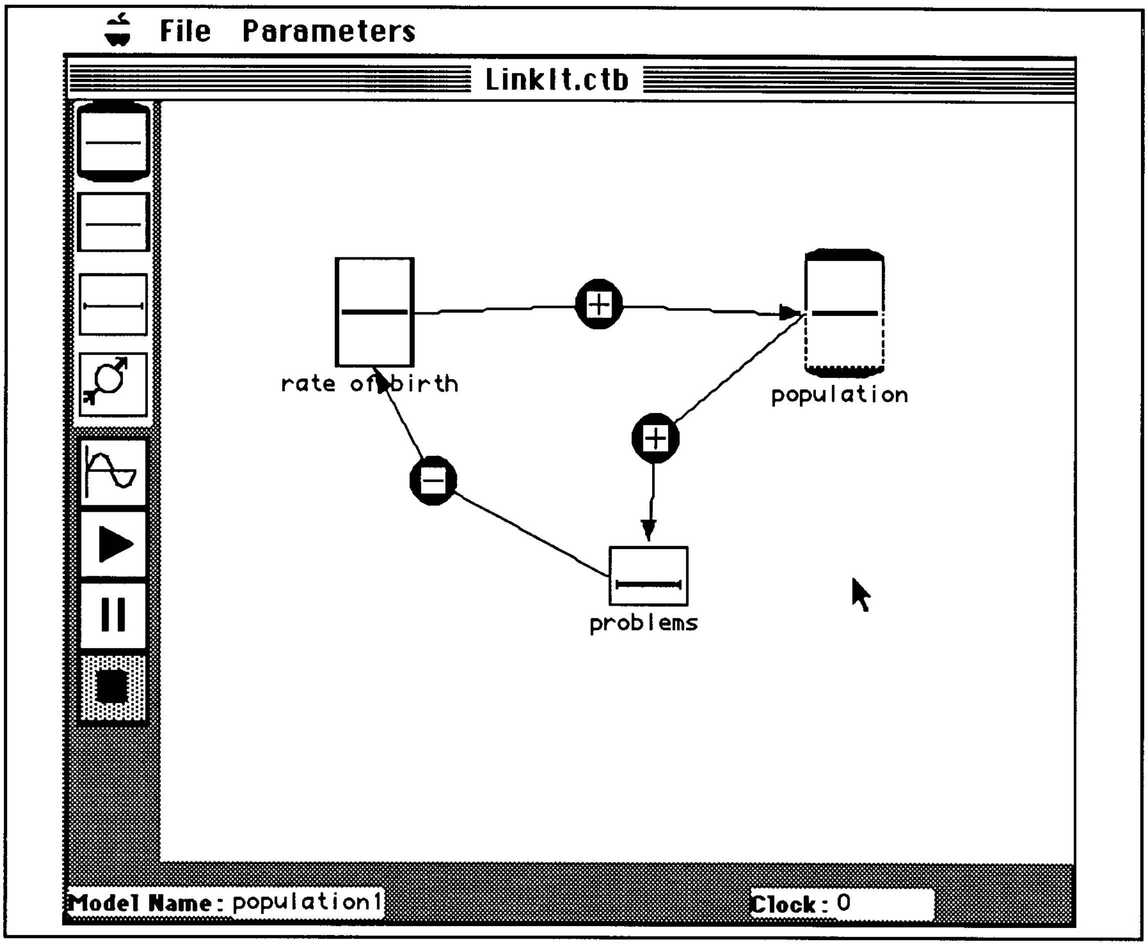


Figure 4.1: A simple model with LinkIt: The screen objects rate of birth, population and problems are known as LinkIt's variables. The arrows between them are known as links.

The central idea with this style of interface is to try to focus the attention of the user on the objects and actions of interest, replacing, for instance, a command-based interface where the operations are made through a more artificial and indirect process (Shneiderman, 1992).

To create a model with LinkIt, the user has to define the variables and relationships and attach to them some characteristics that reflect the functional aspects of the objects and events being modelled. The system, in its turn, is responsible for running the model. In order to do so, it executes two steps: First it has to choose a set of mathematical equations (always hidden from the user) - based on the characteristics of the variables and relationships already created - that will control the behaviour of the model. Second, it makes the model evolve over time through the calculation and iteration of these equations. During this process of running the model, the user receives feedback from the system, seeing changes in some physical aspects of the objects on the screen.

4.4 LINKIT: SUBSET OF SYSTEM DYNAMICS MODELLING THAT IT CAN REPRESENT

LinkIt provides a constrained control structure that can be used to define how one factor affects another. These structures, together with their properties, were carefully chosen to provide to the user a basic set of building blocks to construct models without the overhead of learning formal mathematics or a computer language.

Basically the system provides three different control structures:

- *Go together structures* - They are used to define relationships where the value of the affected factor is immediately calculated based on the value of the causal factor. Mathematically it can be said that this structure serves to represents relationship of the kind $y = a*x$, where a is a constant that can be modified by the user;
- *Cumulative structures* - They are used to define relationships where the value of the causal factor can be seen as a rate of change of the dependent factor. Mathematically it can be said that this structure serves to represent relationship of the kind $y(t+1) = y(t) + a*x$, which is a discrete time step approximation of the linear differential equation $dy/dt = a*x$, where a is a constant;
- *On/Off structures* - They are used to represent situations where it is necessary to determine when a certain event will happen. It is important to mention that the triggering of the event is not based on the system clock but on the value of the causal factor(s).

These three elements of the system can be combined using a fourth component - link - which serves to specify a relation between them (Section 4.5.3.3 presents the modelling possibilitites of LinkIt I and some examples).

4.5 DESCRIPTION OF THE COMPONENTS OF THE SYSTEM

4.5.1 The Desktop

The metaphor used by the system is that of a "blank working sheet" where the user can "draw" a model and see it evolving with time. So, when the system is loaded a window is automatically created and presented to the user (see Figure 4.2).

This window is divided into two areas: the *system area* is the place where the control panel, control fields and the name (version) of the software are presented. It has more or less the appearance of a frame. The *working area* is the place where the models are created or loaded. It is the white part of the window, very much like a blank sheet, dedicated to the construction of the models. Above this window lies the menu bar.

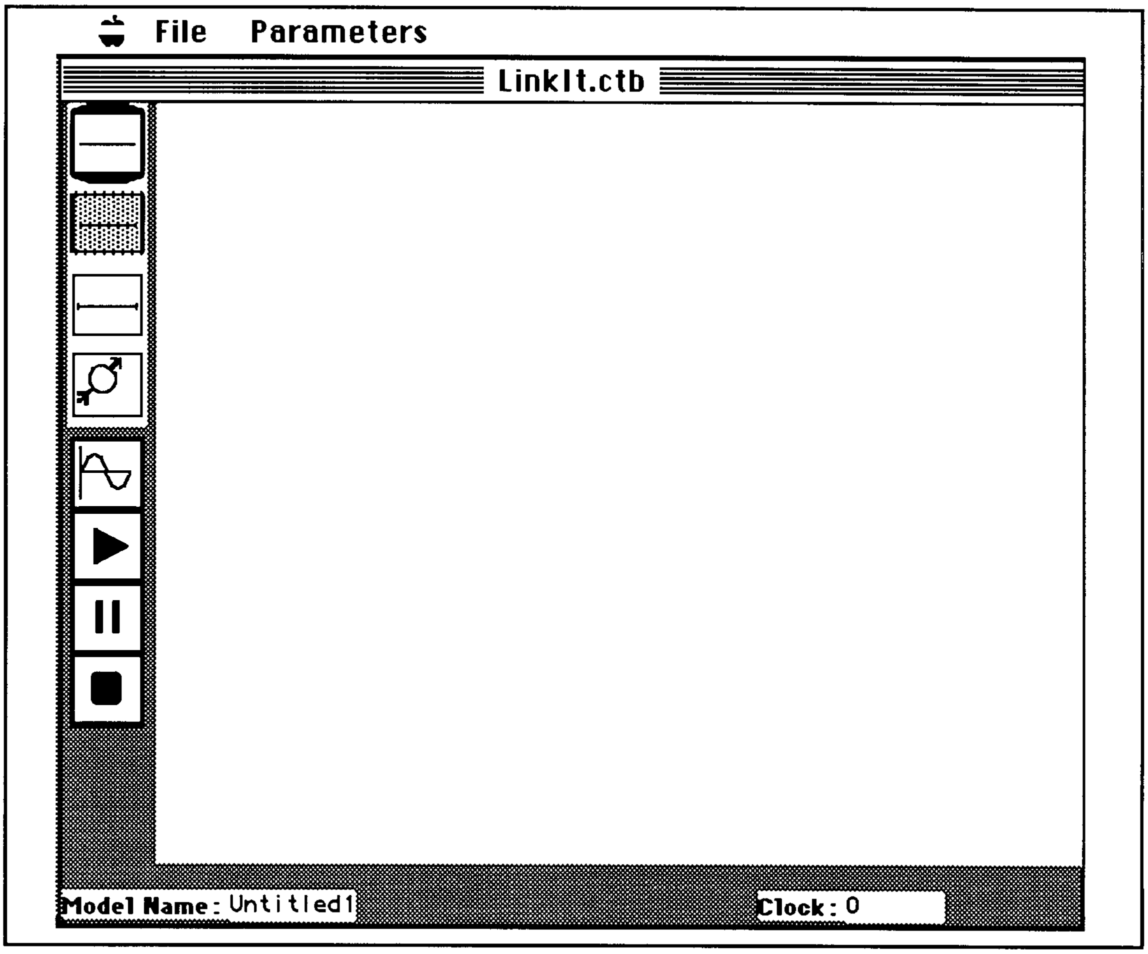


Figure 4.2: How the computer screen looks like when LinkIt is loaded

All components of the system interface are always presented on the screen: A *control panel* located at the left side of the window (see Figures 4.2 and 4.4), contains graphical buttons (icons) responsible for the creation and execution of models. A menu bar located above the top of the window has two "pop-down" menus from where secondary functions like *Save a model* and *Load a model* can be evoked. There is also a *name field*, where the name of the actual model is presented, and a *clock field*, which presents the iterating step when the system is running. Both fields are shown in the bottom part of the window.

To create a model with LinkIt it is just necessary to select the button in the control panel that corresponds to the desired type of object-variable and then to click on the working area. A variable box will be created in that position and the system will ask the user to give a name to it. Links can be created in the same way. The user selects ithe link button on the control panel and then has to inform the system which variable boxes he/she wants to connect by clicking on the variable from which the link is to be made, and then on the variable to which it should go (see Figure 4.3).

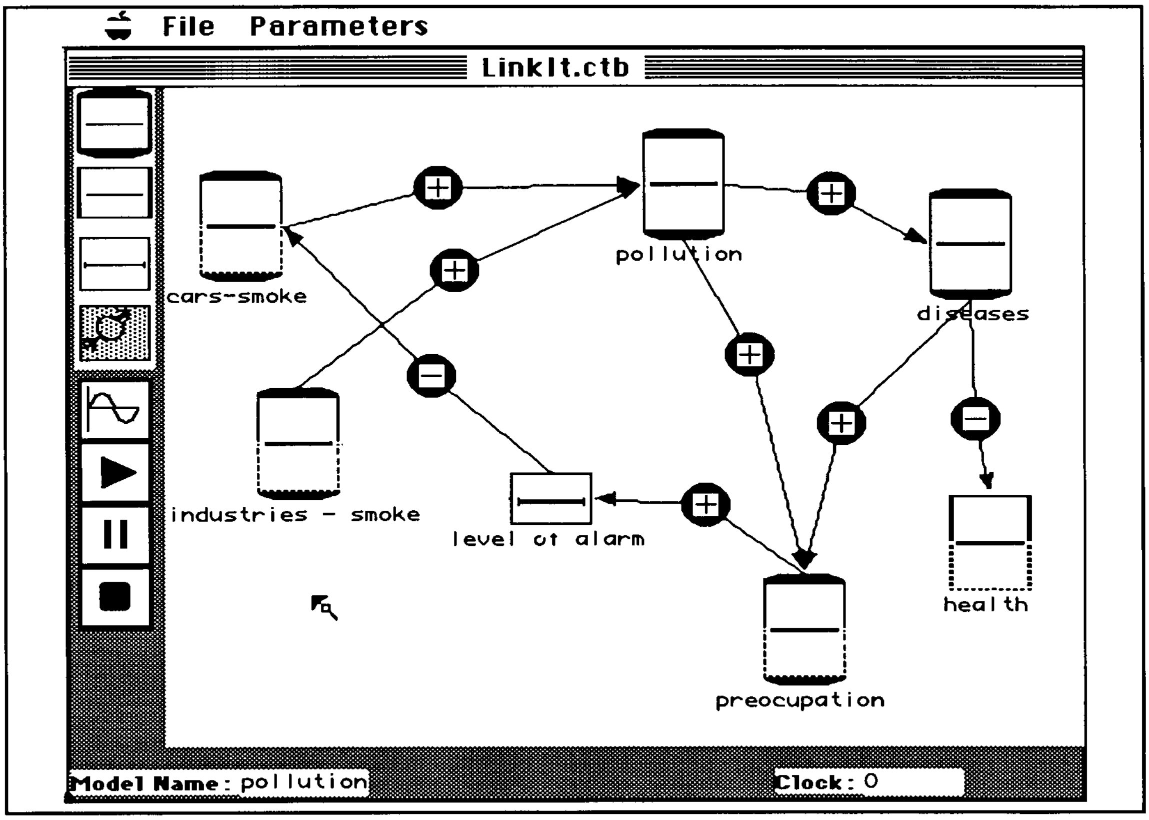


Figure 4.3: LinkIt's window with a model being constructed. The button related to the creation of links is selected on the Control panel. The appearance of the cursor on the screen resembles the operation in progress (create a link). If the user wants to create a link between *level of alarm* and *industries-smoke*, he/she has to click first on the variable box *level of alarm* (the system will highlight it to show that it was selected) and afterwards on the variable *industries-smoke*

4.5.2 Control Panel

The control panel has eight buttons and is located on the left side of the screen (see Figure 4.2 and 4.4). The main reason for positioning it in a vertical way is to leave more space for the working area (most screens are wider than they are tall).

The control panel is divided into two functionally different groups. Each group has a set of basic manipulations (operations) that can be evoked by selecting a button (see Section 4.5.4 for a description of these operations). The first group has 4 buttons and is related to the creation of a model. The buttons there represent the 4 different objects[1] (or building blocks) someone can use to create a model. The second group also has 4 buttons and is related to the simulation of the model already created.

In order to visually differentiate the two sets of buttons, they are presented on the screen with two different sizes over a shaded background. They are also separated by some "blank" lines where only the background is shown.

[1] From the system point of view there are two groups of objects: object-variables (three different types) and object-link.

To select a button the user has to point the mouse to it and single click. When a button is selected, it is automatically highlighted (actually the icon that represents it is filled with a shaded grey pattern) and the previously selected button becomes normal again. In Figure 4.3 the second button from the top is selected and the others are normal.

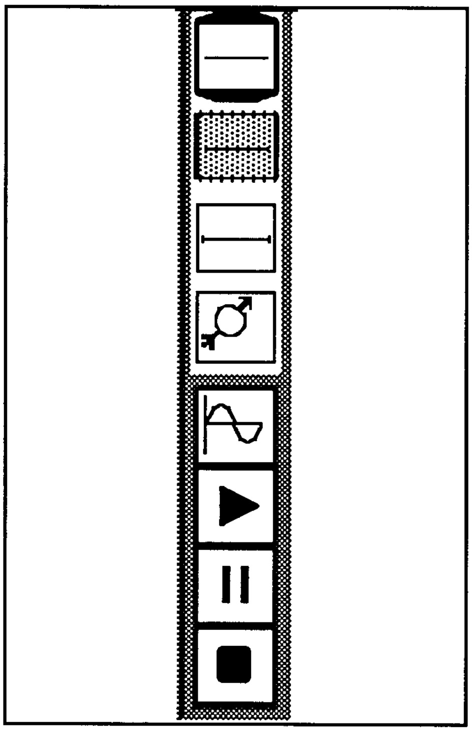


Figure 4.4: Control Panel

4.5.3 Description of the Objects of the System

Causal loop diagrams are the standard way of constructing models under the system dynamics approach, and are a starting point for those interested in quantitative modelling (Roberts, 1983). It is also a very simple representational formalism and yet makes it possible to represent fairly complex problems. These ideas were the key reasons for using that symbolism as the basis for representing cause-effect problems with IQON (see Chapter 2 for a description of the system dynamics approach, causal loop diagrams and IQON). The metaphor used to implement that symbolism was that of direct graph representation. Nodes represent the variables or events of the domain and the arcs (links) represent the relationships between them.

During the design of LinkIt it was decided to follow this same metaphor for representing the models. The reasons are those given above together with the fact that they proved to be fairly intuitive for the pupils involved in the empirical studies with that tool (Bliss, et al., 1992a; Bliss, et al., 1992b; Kurtz dos Santos, 1992).

However some modifications were introduced mainly to extend the possibilities of the system and to make it closer to the modeller's original intentions.

Another important element in causal loop diagrams is the direction of effect of a causal relationship. There a plus ("+") sign represents a positive relationship (the dependent factor follows its cause) and a negative ("-") sign means that the dependent factor goes in the contrary direction of its cause. LinkIt I uses this same notation to represent the direction of effect of their links.

From the point of view of constructing a model, the system has two main building blocks: *object-variables* [2] and *object-links* which can be manipulated by users to externalise their ideas about a certain domain.

These two elements are presented and discussed in the next two subsections.

4.5.3.1 Object-Variables and their Information Boxes

This prototype has three different types of object-variables that can be used to represent variables or events of the domain. Each object-variable has a set of properties (or attributes) that can be used to adapt them to the problem the user wants to model.

These different types of variables and their properties extend the categories of problems that could be modelled with IQON.

An object-variable is represented on the screen by a box (also called a variable-box) that shows its most important properties. Their other attributes can be viewed (and changed) by evoking an *Information Box* attached to each of them (the user has to double click on a variable box). Figures 4.7, 4.8 and 4.9 below, show the three different Information boxes related to the three different types of variables.

All variable-boxes have a visible level indicator (inside the box) that can be moved (by dragging) between two extremes (maximum and minimum) either by the user to set an initial condition to that variable or by the system when it is running the model (in this case only with 'gradual' and 'immediate' variables). This level is intended to represent an amount of something. For instance, if a variable exists called *living conditions* (in a certain country), the more to the top the level is, the better the living conditions are in that country. The more to the bottom the level is, the worse the living conditions are. When the level is in the middle it can be interpreted either as normal or zero (see also section 4.5.5).

Two choices had to be made about the variation of the level. One was concerned with its direction of variation (horizontal or vertical) and the other was about how to represent quantities.

---

[2] Object variables can be of 3 types, representing the 3 different control structures described in section 4.4.

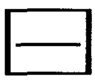
The choice of variation of the level in a vertical direction (instead of horizontal variation) is based on the fact that the idea of bigger values is intuitively associated with the ideas of "being above" or "being in an upper level".

There are two main choices for representing the quantity associated with a given variable. The first follows the STELLA metaphor, where one type of variable represents tanks that can be filled or emptied. In this case, the amount of a variable is visually represented by a shaded area of the corresponding variable box. However, pilot studies made with BOX MODELLER (an ancestor of IQON) suggested that children had a tendency to understand an empty tank as being at zero or rest level and not as at a 'much lower than normal' level (Miller, et al., 1990). The second choice - which is the one chosen for this implementation - uses the actual position of the level in relation to the extremes of the variable box to represent a quantity implied by the variable name. This option has two key elements that very much contributed to its choice. First it is at a higher conceptual level, in comparison to the previous choice, permitting the modelling of a broader number of problems[3]. Second it seems to be intuitive for the user who is accustomed to use/read level indicators in daily experience (Erickson, 1990).

Gradual object-variable

A 'gradual' variable-box is used to represent cumulative values. The variable of this type uses the combined inputs to increase itself a small amount, varying monotonically with the input. When a variable like this is created its amount level indicator, by default, is set to the middle (its initial internal value is zero).

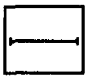
Immediate object-variable



An 'immediate' variable-box is used to represent objects or events that follow its causes. The variable sets its own value to that of the combined inputs. When a variable like this is created its amount level indicator, by default, is set to the middle (its initial internal value is zero).

[3] If we take the example of representing how beautiful a person is, it is better represented by the height of a level than using the idea of a tank filled. This second case is more appropriate to situations where there is an implicit idea of something being accumulated (See also sections 2.5.3 and 2.6.3.1).

GONOGO object-variable



GONOGO variables can be used as an on/off Boolean, switching when its input goes above or below a threshold level indicator which the user can set. When a variable-box like this is created its threshold level indicator, by default, is set to the bottom, meaning that it will be on for any positive value of its input.

The properties of object-variables are:

- *Type* - Defines the type of an object-variable. It can be 'gradual', 'immediate' or 'GONOGO'
- *Level or range* - Defines the maximum and minimum values for 'immediate' and 'gradual' object-variables (see Figure 4.5). It can be *any value* - the internal value of the variable varies in the interval [-1, 1]; *bigger than zero* - the internal value of the variable varies in the interval [0, 1]; *smaller than zero* - the internal value of the variable varies in the interval [0, -1]. The appearance of a variable-box on the screen is related to its range. When it is 'any value', it is represented by a full box. When it is 'bigger than zero' the bottom part of the box is shaded, and the contrary, in the case where it is 'smaller than zero'. 'GONOGO' variables are only 'bigger than zero' and appear on the screen with the size of a half full box with no dashed part.

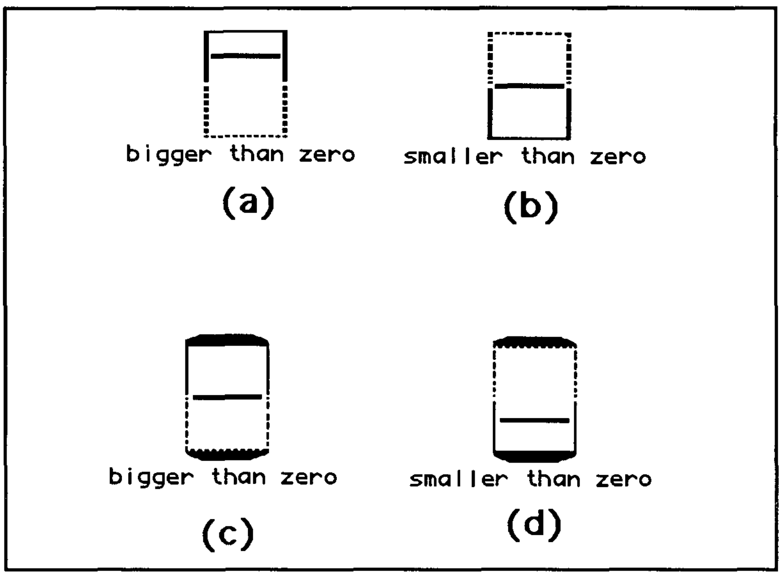


Figure 4.5: (a) An 'immediate' variable 'bigger than zero' (b) An 'immediate' variable 'smaller than zero' (c) A 'gradual' variable 'bigger than zero' (d) A 'gradual' variable 'smaller than zero'

- *Input combination* - The input combination determines how the inputs of a variable will be combined in order to calculate the new value of the variable. These inputs can be combined in three different ways (see Figure 4.6 and 4.8): *Additively* - averaging inputs with weightings for each input (tries to simulate a sum of the inputs). On the Information box it is represented by the toggle button "average"; *And-like* - equal to the smallest weighted input (tries to simulate an AND logical operation). On the Information box it is represented by the toggle button "need all"; *Or-like* - equal to the largest weighted input (tries to simulate an OR logical operation). On the Information box it is represented by the toggle button "need any". (see Section 4.5.5 to know how the system calculates the new value of a variable)

- *State* - This property determines if a variable should be considered in the calculation when the model is running. When a variable is *sleeping* it is not considered in the calculation of the next iteration, but it is considered if it is *awake*. The usefulness of this property is that the user does not need to delete variables if he/she wants to see a model running without them.

- *Output Combination* - This property only exists for 'GONOGO' variables. It determines what will be the output of the variable when it is on. The *maximum* option means that the output will be equal to one (1) regardless of the resultant input of the variable. In the case the option is *same as input*, the resultant input will be copied to the output of the variable.

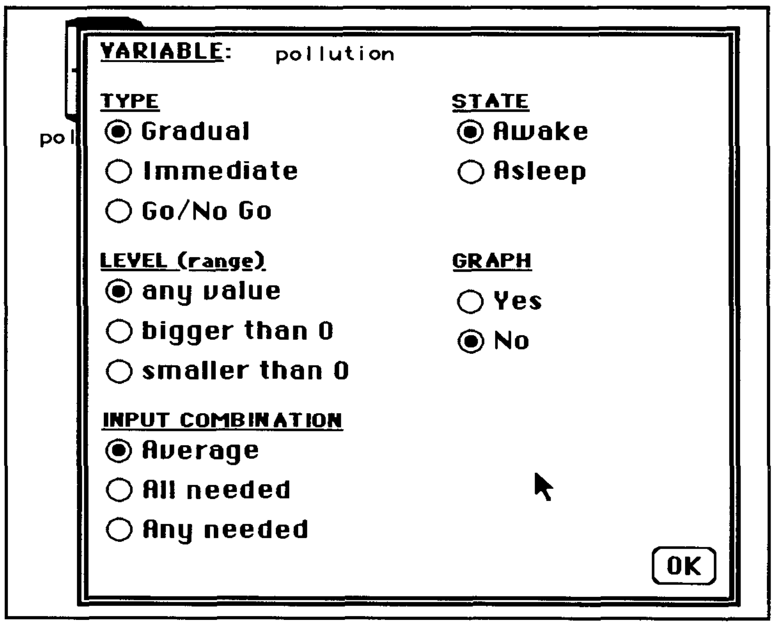


Figure 4.6: Information Box of a 'gradual' Variable

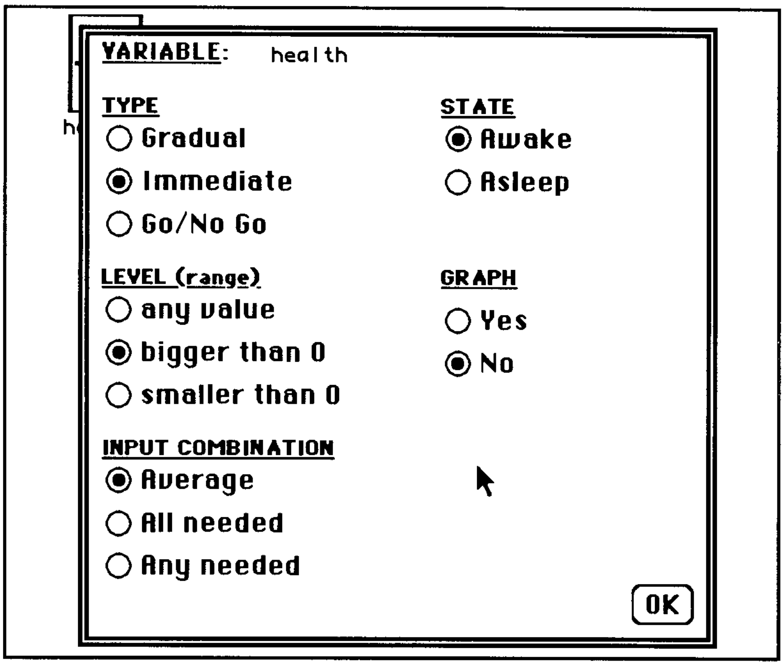


Figure 4.7: Information Box of an 'immediate' Variable

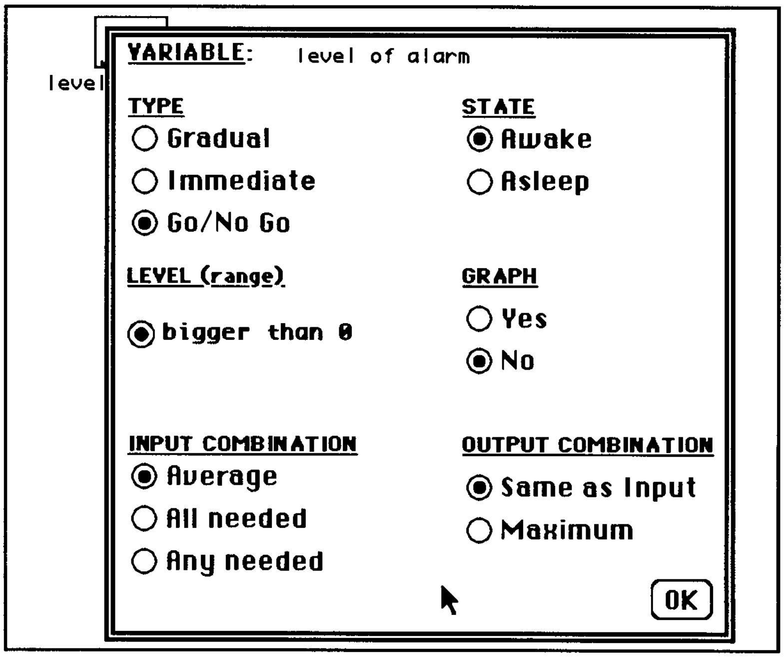


Figure 4.8: Information Box of a 'GONOGO' Variable

4.5.3.2 Object - Link and its Properties

Links are used to create a relationship between variables. Like object-variables, they also have some properties that can be used to adapt them to the problem the user wants to model. In this version there are two different properties: the type of the relationship and its strength (see Figure 4.9).

The type of a relationship can be of two kinds: The *same direction relationship* is used when the user wants the dependent variable to follow the result of its input (e.g. if the input variable is positive it will pass a positive value for the dependent variable). The *opposite direction relationship* is used when the user wants the dependent variable to follow the opposite (contrary) value of the result of its input (e.g. if the input variable is positive it will pass a negative value for the dependent variable). These two different types of links are represented respectively by a "+" sign and a "-" sign inside a black circle in the middle of the link.

The strength of a relationship is also divided into two kinds: The *normal strength* indicates that the weight of a certain link on the calculation of a dependent variable is equal to 1. The *super strength* has a weight equal to 2. So, if a variable called *pollution* has two inputs, say *cars* and *industries*, and they are connected to *pollution* via a 'normal positive' link and a 'super positive' link, respectively, it means that the variable *industries* exert more influence on *pollution* than *cars.* When the strength of a link is 'super plus' or 'super minus', the sign in the middle of the link is shown in bold face.

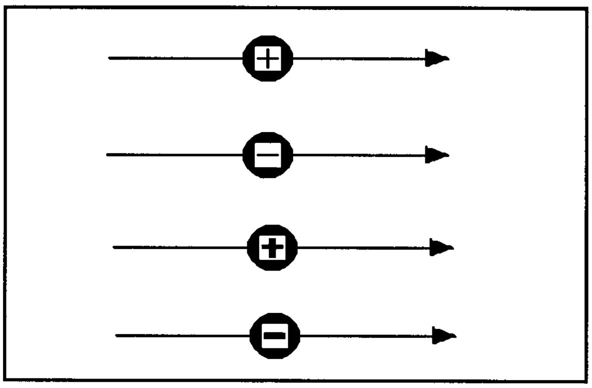


Figure 4.9: Links: Types and Strengths

4.5.3.3 Modelling Possibilitites and Examples

LinkIt can be used to construct models which bear a resemblance to a number of classes of mathematical models. Broadly, the 'immediate' variables are related to algebraic models, the 'cumulative' variables to differential equations, and the 'GONOGO' variables to boolean models. Examples of these, alone and in combination, will be given below.

<u>Algebraic models</u>

Models in an algebraic style can be created by using exclusively 'immediate' variables. Their resemblance to true algebraic models is constrained by the three different ways of combining links. Using the 'average' input combination, it is possible to represent something like linear models of the form (see Figure 4.10.a):

$y = a^*x_1 + b^*x_2 + ...$

Using the 'need all' combination it is possible to obtain a qualitative behaviour that resembles the mathematical operation multiplication (at least in the sense that the result is zero if any of the input variables is zero). Thus, models of the form

$$y = K^*x_1^*x_2 \ ...$$

can be represented to a limited extent (see Figure 4.10.b).

The 'need any' combination also serves to represent additive models, without the averaging feature (see Figure 4.10.c).

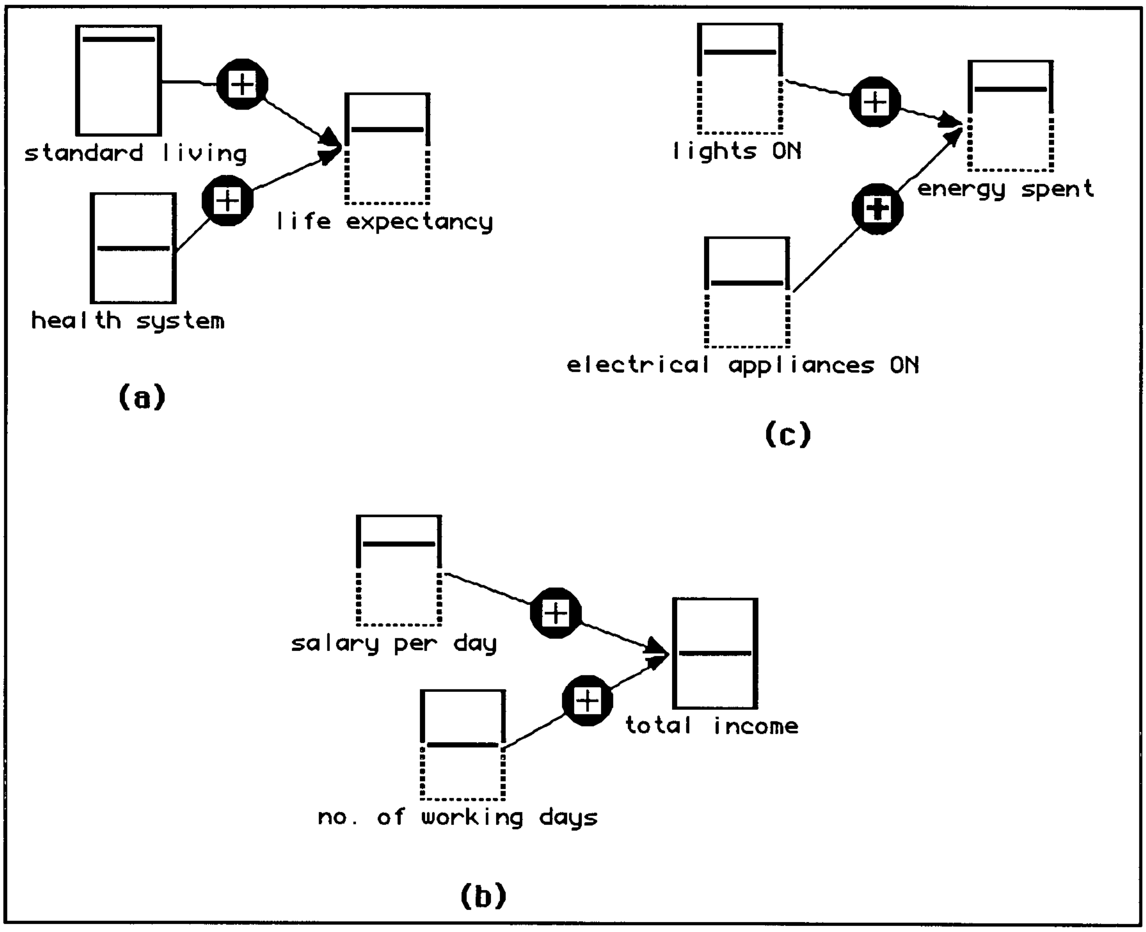


Figure 4.10: Three examples of algebraic models: (a) Model about life expectancy using 'average' combination;. (b) Model to calculate the total income of a worker based on the salary paid per day and the number of working days in a month (use of 'need all' as an approximation to multiplication); and. (c) Model to calculate the energy spent by the light bulbs and the electrical appliances that are ON (use of 'need any' as an approximation to addition)

Boolean Models

The 'GONOGO' variables can be used to construct boolean models of the kind:

$$y = IF\ (X_1\ AND\ X_2\ AND\ ...)\ THEN\ 1\quad ELSE\ 0$$

or

$$y = IF\ (X_1\ OR\ X_2\ OR\ ...)\ THEN\ 1\quad ELSE\ 0$$

The system is not really designed to construct pure boolean models, but rather to allow boolean dependencies within other models. Figure 4.14 shows an example of model using a 'GONOGO' variable.

Differential Equations

The 'cumulative' variables with one or more fixed independent variables as inputs, provides an approximation to differential equation of the form:

$$dy/dt = K$$

where K can be positive or negative, and may be obtained from different combinations of inputs (e.g., an 'average' of several inputs). Thus the system provides for modelling representations of essentially linear increase and decrease.

The graphical solutions for this class of problems have the following form (Figure 4.11):

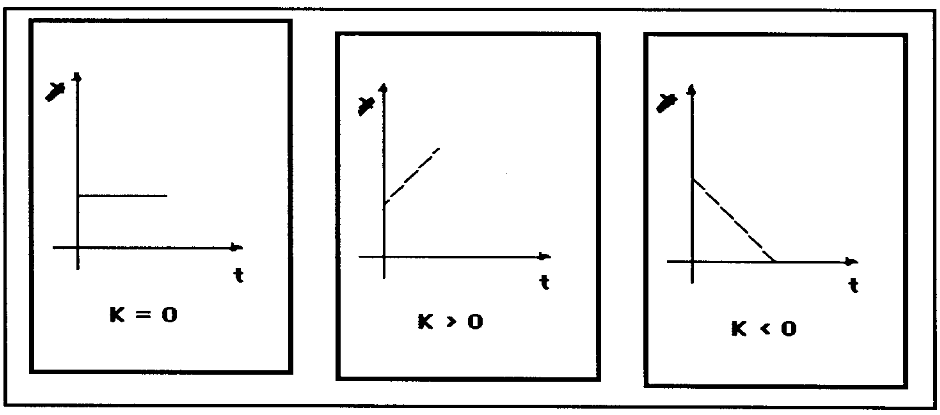


Figure 4.11: Linear Processes - Graphical solutions

Figure 4.12 shows an example of a problem within this category modelled with LinkIt. The variables *salary* and *Balance of my Account* can be defined as:

Salary = Inflation

d(Balance of my Account)/**dt** = Salary

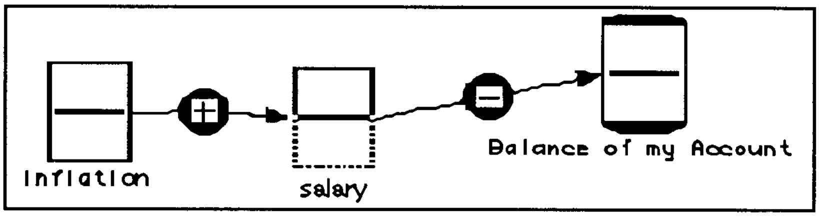


Figure 4.12: Examples of a linear process modelled with LinkIt I

It could be thought that *salary* was defined as 'bigger than zero' because it makes no sense of receiving a salary less than zero. However, *Inflation* and *Balance of my account* can assume negative values if someone thinks about deflation and overdraft.

By using additional 'immediate' variables, the value of a 'cumulative' variable can influence itself, leading to exponential models of the form: $dy/dt = K*y$

The possible graphical solutions for this category of problems is shown in Figure 4.13.

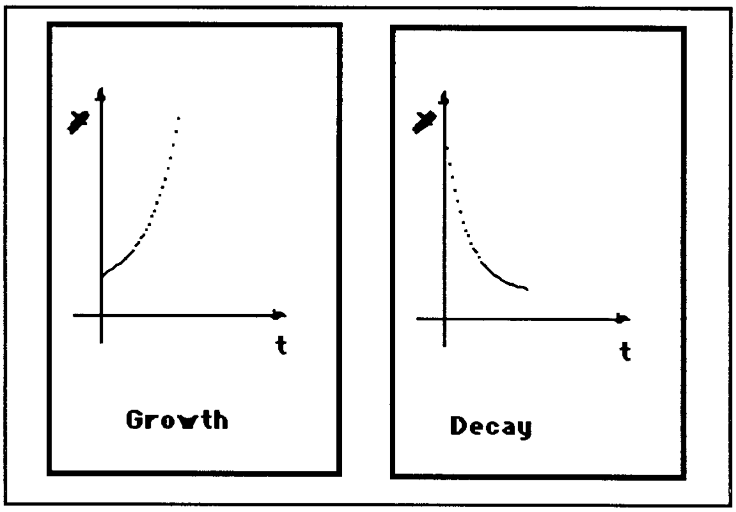


Figure 4.13: Exponential growth/decay - graphical solutions

Figure 4.14 shows two examples of problems within this category modelled with LinkIt. In both examples *population* was defined as 'bigger than zero' because it makes no sense to have a negative population.

The first example was expanded to represent the idea of the social problems starting after a certain level of population is reached. In LinkIt this idea can be represented by a 'GONOGO' variable (*overpopulation*) and an 'immediate' variable (*social problems*).

The second example is a case of exponential decay.

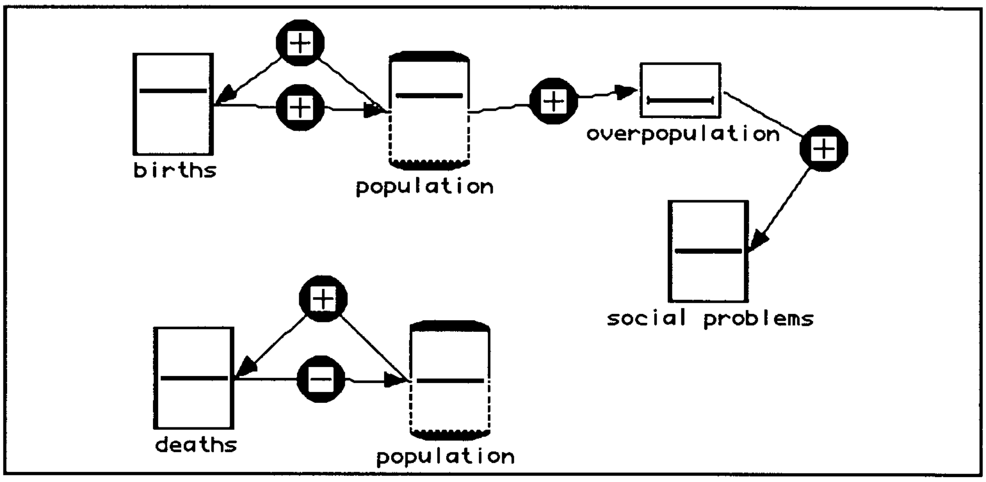


Figure 4.14: Examples of exponential growth and decay modelled with LinkIt I

These two models can be combined to represent in a more realistic way the problem about population (see Figure 4.15). It is mathematically represented by the following equation (*b* and *d* are respectively the percentages of birth and death)[4]:

$$\frac{dPop.}{dt} = (b - d)*Pop.$$

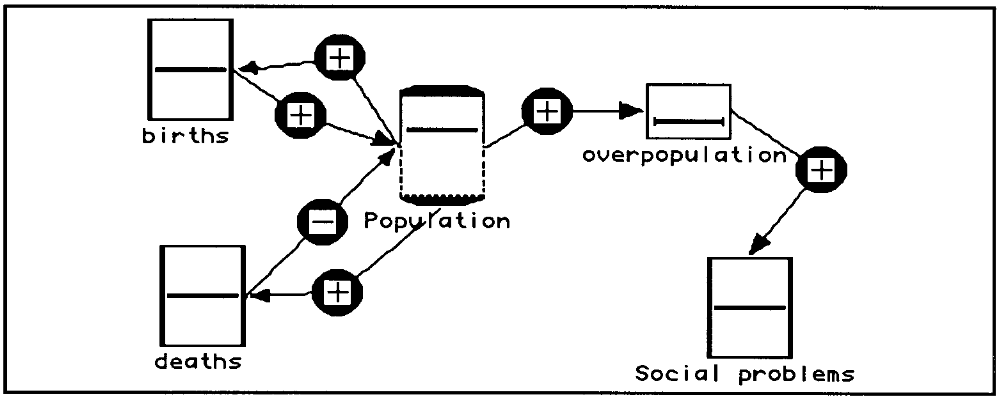


Figure 4.15: Model about population - Combining exponential growth and decay

Another way of achieving exponential decay with LinkIt is using the damping value (see topic "Menu Parameters" in Section 4.5.4.3 for a description of the damping function). Figure 4.16 below shows an example of its use in a model about a "leaky tank".

[4]The mathematics of the system imposes two restrictions in respect to this problem. First is the fact that the constants *b* and *d* are averaged. Second is that the system provides only 2 possibilitites for the values of *b* and *d* through the different strengths of the links.

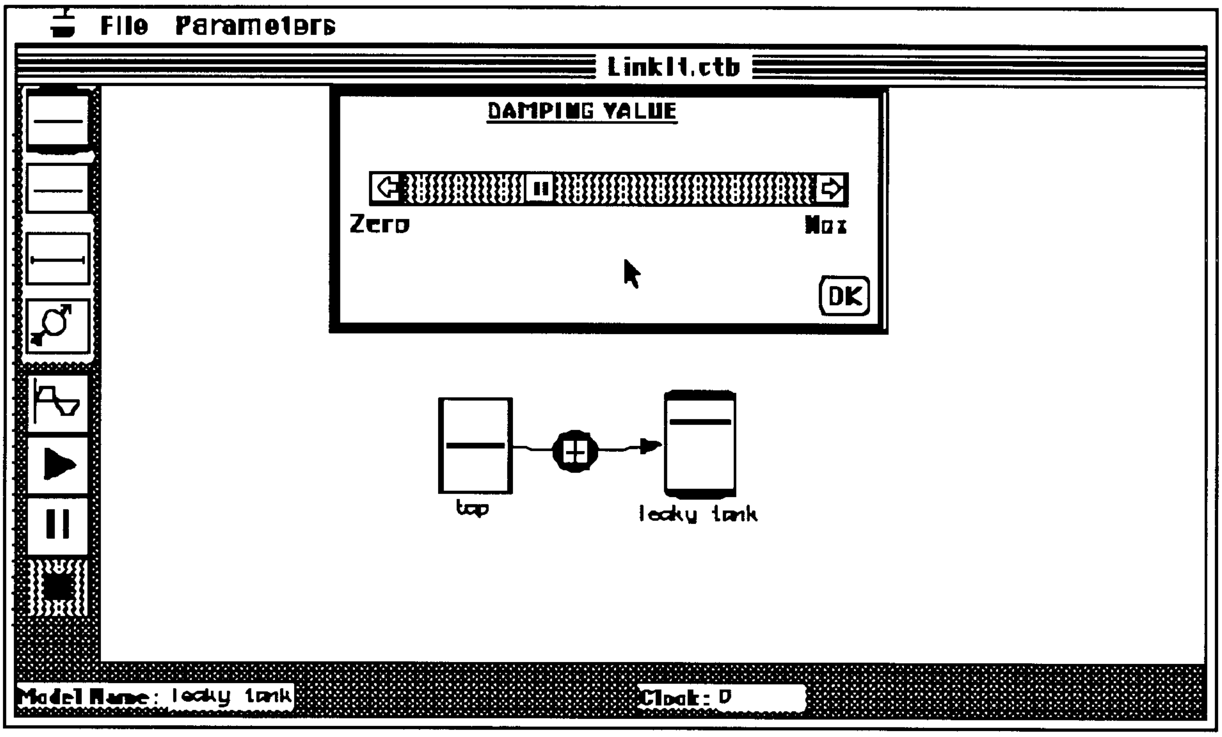


Figure 4.16: Example of an exponential decay model using the 'damping value'

Second or higher order derivatives can be represented approximately by composing first order 'cumulative' variables. The example shown in Figure 4.17 illustrates the classical example of mechanics where

d(displacement)/**dt** = velocity
d(velocity)/**dt** = acceleration

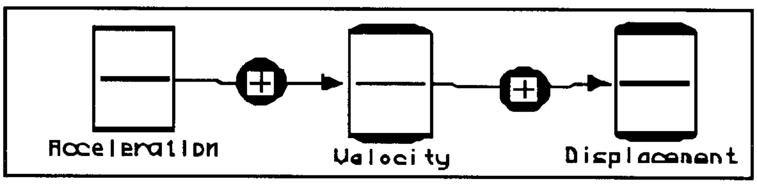


Figure 4.17: Example of a second order derivative model

Another important class of problems related to second order derivatives is oscillation, which are mathematically represented by the differential equation:

$$m * \frac{d^2x}{dt^2} + b*\frac{dx}{dt} + K*x = 0$$

A typical example within this category is the "predator-prey" problem. Figure 4.18 shows a model constructed with LinkIt to represent this problem and Figure 4.19 presents its graphical solution (showing the graph of *prey*).

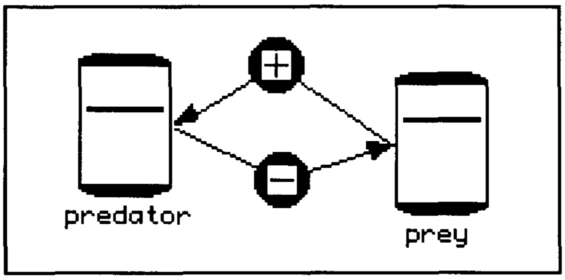


Figure 4.18: Example of oscillation with LinkIt I

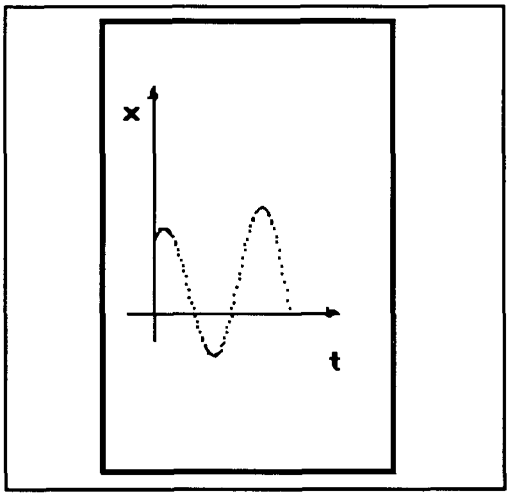


Figure 4.19: Possible graphical solution for oscillation-type problems

Combined Models

'Immediate' and 'GONOGO' can be used in combination with 'cumulative' variables. An example is shown in Figure 4.15 in which *births* and *deaths* are combined (in effect by subtraction) to determine the growth of a population, which if it becomes bigger than a certain predetermined value (the value indicated by the level of *overpopulation*), generates social problems.

It is worth mentioning at this point that LinkIt II removes several of the restrictions and limitations on the kinds of models discussed above, bringing them closer to orthodox mathematical forms (see Section 6.2.1.3).

4.5.4 Operating the System

LinkIt provides three groups of interface operations (see Chart 4.1). The first group - edit operations - is related to creation/modification of a model. The second - running operations - is concerned with the iteration (or running) of a model which already exists (the two sets of graphical buttons in the control panel are arranged according to these groups). The third group - secondary operations - is concerned with the model as a whole and is evoked via pop-down menus.

4.5.4.1 Description of the Editing Operations

Before beginning to describe these interface operations, it is important to introduce the idea of *hot region.* A *hot region* is the part of an object on the screen that is sensitive to the manipulations made by the user. When the object is a variable-box, this region includes the whole rectangle demarcated by its boundary. Variable-boxes have also an *embedded hot region* which is the rectangle that surrounds the amount/threshold level indicator of the variable. The *embedded hot region* has a high priority, which means that if the user clicks and drags on the region near to the level indicator, the system will interpret the action as " move the level" instead of "move the variable-box". If the object is a link, its hot region is the black circle that is shown in the middle of the link.

<u>Create an Object</u>

To create a new object (variable or link), the user has to select the corresponding button from the control panel (by single clicking on it) and to single click (again) on the working area where he/she wants to place the new object. If the required object is a link, after selecting the appropriate button in the control panel, the user has to single click on the two variable-boxes he/she wants to connect. If the system does not accept the operation, no object is created and the system remains in that mode of operation. There are two exceptions to this rule. The first exception is when the user tries to connect a variable to itself. In this case the system presents a warning box telling the user that self-connection is not allowed. The second case is when the user clicks on another button on the Control panel. In this case the system assumes that the user wants to invoke another Edit/Running operation, entering into the mode associated with that new operation.

Create objects operations have a higher level of priority over the *basic manipulations* in such a way that once one of these buttons is selected, the system enters in a mode of operation associated with that button and whatever other basic manipulation the user does, the system comes back to the state of creating an object. This means that the user does not have to keep clicking on the buttons associated with the objects, if he/she wants to continue performing the same operation.

When a button to create an object is selected, the cursor shape changes its appearance, resembling the object selected. The idea here is to give the user further visual reinforcement about what object is selected.

Basic manipulations, particularly moving a box or link, or changing a level, permit the user to interact with the objects on the screen without the necessity of selecting a button to perform them. In this sense they are implicit operations and can be performed at any time by the user.

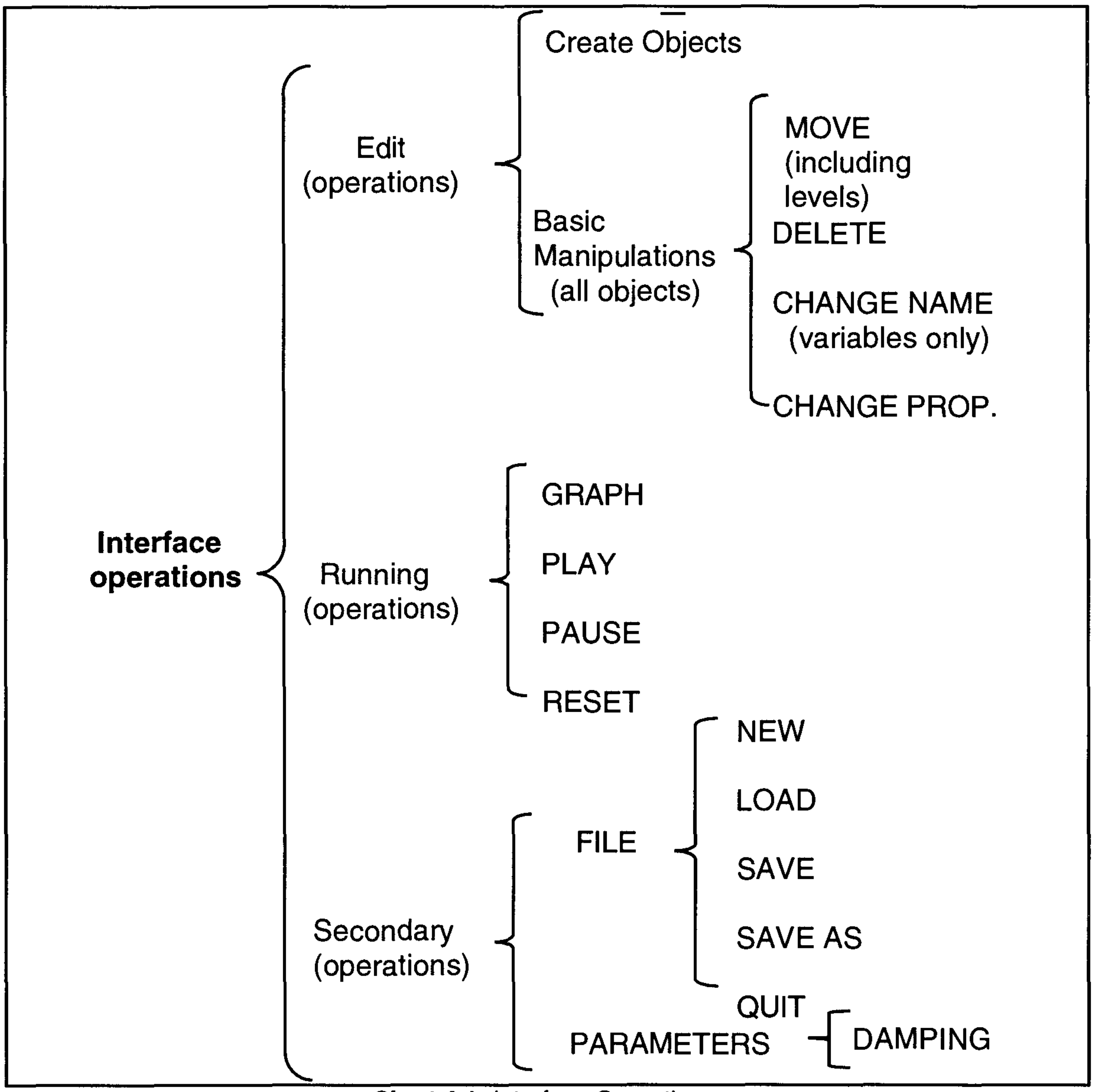


Chart 4.1: Interface Operations

<u>Basic Manipulation - Move an Object (Variables and Link)</u>

To move an object from its actual position it is necessary to click on it and drag it to its new position. If the object is dragged to a region outside the working area, the system will try to redraw it in a position close to the border which the user tried to move beyond.

During the process of moving an object the system gives real time feedback of the new position, by presenting a dashed outline copy of the object being moved.

Basic Manipulation - Move the threshold/amount Level indicator

To move the level indicator of a variable, the user has to click and drag it up or down. As the mouse is moved the level indicator goes with it in real time. When the level arrives at either extreme of the box it stops moving (even if the user continues to move the mouse).

Basic Manipulation - Delete an object

To delete an object the user has to select the object by clicking on it and pressing the delete key. If the object is a variable all links that depart from and arrive to that variable are automatically deleted by the system.

Basic Manipulation - Change the name

To change the name of a variable, the user has to single click anywhere on the name . An editing box is automatically shown and the user can edit the new name in a way very similar to the editing mode of a word processor. The system will accept the new name after pressing the return key.

Basic Manipulation - Change the properties (object-variable)

Object variables in LinkIt have a set of properties that can be changed by the user. They appear in an Information box that appears when the user double clicks on a variable box. All of these properties are presented as toggle buttons in this Information Box.

Because not all properties apply to all variables, there are two different Information boxes (see Figures 4.6 and 4.8), one for 'gradual' and 'immediate' variables and another one for 'GONOGO' variables. If the user changes the type of a variable, the Information box is automatically reformatted to be in correspondence with the new variable type.

After changing the properties of a variable, the user has to click on the 'OK' button in the bottom part of the Information box in order for the system to modify the visual appearance of the variable (if necessary) and proceed.

Basic Manipulation - Change the properties (object-link)

As described in topic 4.5.3.2, the properties of a link are its type and strength (see Figure 4.9). In order to change them, the user has to double click on the black circle in the middle of the link. The two types and two strengths are in a sequential order. Each double click makes the system present the next possibility in the following order: 'plus', 'minus', 'super plus' and 'super minus'.

4.5.4.2 Description of the Running Operations (Related to the Execution of the Model)

To execute any running operation the user simply has to single click on the corresponding button in the control panel. When a button is selected it becomes highlighted.

To move back to any edit operation the user simply has to click anywhere in the working area (e.g. try to perform a basic manipulation) or on one of the buttons on the control panel that correspond to the creation of objects.

More details about each of the running operations are given below.

Graph

This button permits the graph of a certain variable to be shown when the model is running. After selecting the button, the user has to inform the system which variable's graph he/she wants to see, by clicking on it[5]. The graph presented is the variation of the selected variable (y axis) versus time (iterations - x axis).

Run

This button starts the running process. In order to do so, it performs the following steps:

1 - Gets the current values of all variables

2 - Calculates the next new value for each one

3 - Update the values of all variables

4 - Updates all variables on the screen

- 4.1 - Recalculates the new position of the amount level indicator of all 'gradual' and 'immediate' variables and redraws and verifies which 'GONOGO' variables have to become "on" and "off"
- 4.2 - Redraws all amount level indicators and the 'GONOGO' variables

5 - Updates the clock

6 - Checks if there is any event. If yes, stops the running process and goes to treat the event. If not, goes back to step 1.

---

[5] In the beginning of the design of the system it was intended to give the user the possibility of seeing the graph of more than one variable (at the same time) while the model was running. In order to do so the user had to inform the system by setting the graph option on the Information box of the variable (See Figures 4.6, 4.7, 4.8). Later it was noticed that if the user set the graph option for many variables, it would become difficult to accomodate many small windows inside the (small) screen area. So it was decided to leave the option there for future testing when bigger screens were available and during the Introductory study the students were informed that this option on the Information box was not available.

Pause

This button pauses the model when it is running. It can be used to change the value of a variable and continue running the model, or just as a pause to see what is going on with the model. It does not change either the values of the variables or the clock. It is implemented as a toggle button. The first click on it pauses the running process, the second click makes it resume.

Reset

This button stops the running process. It performs the following steps:
1 - Stop the running process
2 - Reinitialises all values of the variables (changes them to their default values) and updates them on the screen
3 - Makes the clock equal to zero and update its field on the screen.

If the model is not running it just reinitialises the values and the clock (performs steps 2 and 3).

4.5.4.3 Description of the Secondary Operations

LinkIt has two groups of secondary operations: *File* and *Parameters*. They are presented to the user via two pop down menus in the menu bar and they are concerned with the model as a whole.

<u>Menu File</u>

The menu File contains functions related to the creation and storage of models (see Chart 4.1). These functions are:
- *New model* - Creates a new model clearing all the workspace area. If there is an unsaved model on the screen, the system will ask the user if he/she wants to save the model first. All internal variables and structures of the system are reinitialised.
- *Load model* - Loads a model previously saved. A dialogue box is shown to the user in order to select the folder and file to be loaded. If the file is not a LinkIt file type, an error message is shown and the system comes back to its previously state.
- *Save model* - Saves the model existing in the working area with the name shown in the name field. If a file with the same name already exists on the disk, it will be

destroyed and a new one will be created. In the case the model has been altered since the last saving, the system performs a "save as" function.
- *Save as* - Saves the model existing in the working area. Before the model is saved, the system opens a dialogue box asking the user to give a new name to the model. After saving the model, the name field presents the new name.
- *Quit* - Quits the system. If there is a model on the working area and it has been altered since the last save, the system tries to perform a "save as" function before quitting.

Menu Parameters

In this version the pop down menu *parameters* has just a damping option. This damping function attaches a natural decay towards the neutral value to all 'gradual' and 'immediate' variables of the model. The intensity of decay can be regulated by the user through a slide bar presented to him/her when the function is evoked (see Figure 4.20). The default value is zero (no decay).

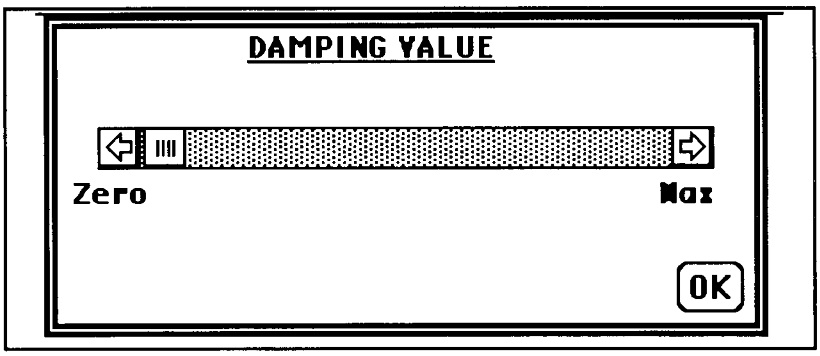


Figure 4.20: Slide bar to set the 'damping value'

4.5.5 The Underlying Mathematics of the Software

The computational basis of LinkIt is based on some properties of artificial neural networks. The first property is that the response of artificial neurons to a certain input is within a certain interval (Miller, et al., 1990; Rumelhart & McLelland, 1987). Therefore their outputs can be seen as "big" or "small". They also have a resting level where no output is generated. Another important feature is that they respond to their inputs in an uniform way through the sum of weighted values[6].

The computational mathematics of LinkIt was developed in such a way that the possible internal values for its variables are within the interval [-1, 1]. So the minimum possible value for a variable is -1, 0 is the 'normal' value and +1 the

[6] It is important to say that these characteristics used in LinkIt does not make it a "conectionist tool" since those systems have a lot of other important properties that are not present here.

maximum possible value. This range of values serves well to represent semi-quantitative values such as "big", "small", "less than enough", "normal", etc. To display values on the screen it is necessary to scale them. The approach taken here is different from STELLA where the user explicitly has to set the scale before any simulation can be seen. LinkIt automatically gives all variables the same scale which fits in the interval [-1, 1]. To keep the variables into this range the system needs to squash the values into it. The strategy adopted here was to use a squashing function sensitive to the range of the variable and to its new calculated input (see Section 4.5.5.1 for how the system calculates the function S(x)).

The internal values of the variables are shown to the outside world via the amount/threshold level indicator using an offset function related to the screen co-ordinates of a certain variable box. So whenever a level is moved (either by the system or by the user) its new external and internal values have to be recalculated.

The computational mathematics for each type of variable of the system is presented below.

Notation used

If we consider that a certain model has n variables with values $a_j (1 <= j <= n)$ and L links (L >= 0) with weights w, then a link between a_i and a_j can be represented by w_{ij}. Let $a_j(t)$ be the value of a variable j at time t and S the squashing function which restricts the value of a certain variable (after recalculating its new value) to the interval [-1, 1].

The calculated value for the system variable **Input** is related to the Input combination chosen by the user for a given object-variable and its m ingoing links. In the case when the input combination is *average* , the value of **Input** (for a certain variable a_j) will be:

$$\textbf{Input} := \left(\sum_{i=1}^{m} (a_i(t-1) * w_{ij}) / \sum_{i=1}^{m} w_{ij}\right) \quad \text{with } a_i \text{ fixed at time t-1}$$

In the case when the input combination is *need all*, the value of Input will be:

$$\textbf{Input} := \left(\underset{i=1}{\overset{m}{\text{Minimum}}} (a_i(t-1) * w_{ij}\right) \quad \text{with } a_i \text{ fixed at time t-1}$$

In the case when it is *need any*, the value of Input will be:

$$\textbf{Input} := \left(\underset{i=1}{\overset{m}{\text{Maximum}}}\ (a_i(t-1) * w_{ij})\right) \quad \text{with } a_i \text{ fixed at time t-1}$$

Note that provided all values a_i are in the range [-1, +1], the value of **Input** will also be in this range.

4.5.5.1 The Running Process

LinkIt runs a dynamic simulation based on an iterative, incremental computational model of the equations (1), (2) and (3) shown below.

After the user sets the model to run and before the first iteration (t=0), the system assumes the current values of the variables as their initial values. From that point the system begins to compute the equations (1), (2) and (3) below for each appropriate variable at each iteration, using the values of the variables at the previous iteration. After completing the calculations for one iteration, it updates the variables (and the clock) on the screen and goes to check if any interruption occurred.

In order to optimise the running process, the system creates two matrices at the beginning of a simulation that contain information about the relationships between the variables and their values. Only 'awake' variables that have at least one 'awake' ingoing link or a damping value attached to them, are present in these matrices.

Gradual variables

Gradual variables have an amount proportional to **Input** added to their current value. They can therefore go outside the range [-1, +1]. To present this, the squashing function **S** (a logistic function) is applied to the increment.

• Calculating the constants:

DT := 0.1 {Constant - Time interval}

Damping := Constant * a_j (t-1) {Damping is a value associated to all a_j and can be set by the user before running a model. It is used to implement a natural decay to a_j }

• Handling the Inputs:

```
If  -1 <= aj <= +1                    { aj is an any value variable }
        then if  Input > 0  then S := 1 - aj(t-1)
        else S := 1 + aj(t-1)         { Input <= 0 }
endif

If  -1 <= aj <= 0                     { aj is a smaller than zero variable }
        then if Input > 0 then S := aj(t-1)
        else S := 1 + aj(t-1)         { Input <= 0 }
endif

If  0 <= aj <= 1                      { aj is a bigger than zero variable }
        then if Input < 0 then S := aj(t-1)
        else S := 1 - aj(t-1)         { Input >= 0 }
endif
```

• Calculating the next value:

$$a_j(t) := a_j(t-1) + (DT * \mathbf{Inpu} \quad - \mathbf{Damping}) * S \quad (1)$$

<u>Immediate variables</u>

By definition 'immediate' variables have no damping associated with them. Also because its next value is a function of its **Input** only - which is always in the interval [-1, 1] - the next calculated value for a certain 'immediate' variable a_j can be determined by its range and the value of its **Input**.

• Calculating the next value:

If a_j is 'any value' then $a_j(t) :=$ **Input** (2.1)

If a_j is 'bigger than zero' and **Input** >= 0
then $a_j(t) :=$ **Input** (2.2)
If a_j is 'bigger than zero' and **Input** < 0
then $a_j(t) := 0$ (2.3)

If a_j is 'smaller than zero' and **Input** >= 0 then $a_j(t) := 0$ (2.4)
If a_j is 'smaller than zero' and **Input** < 0 then $a_j(t) :=$ **Input** (2.5)

GONOGO variables

The calculation of the new value of 'GONOGO' variables is done in a very similar way to 'immediate' variables. The differences are only that the range of 'GONOGO' variables is always 'bigger than zero' and the value of Input is compared to the value of the threshold of the variable (and not to its amount value).

• Calculating the next value:

$a_j(t) :=$ **Input** if **Input** $>$ Threshold(a_j) (3.1)

$a_j(t) :=$ null if **Input** $<=$ Threshold(a_j) (3.2)

4.5.5.2 An Example

Suppose someone created the model shown in Figure 4.21 and is about to run it:

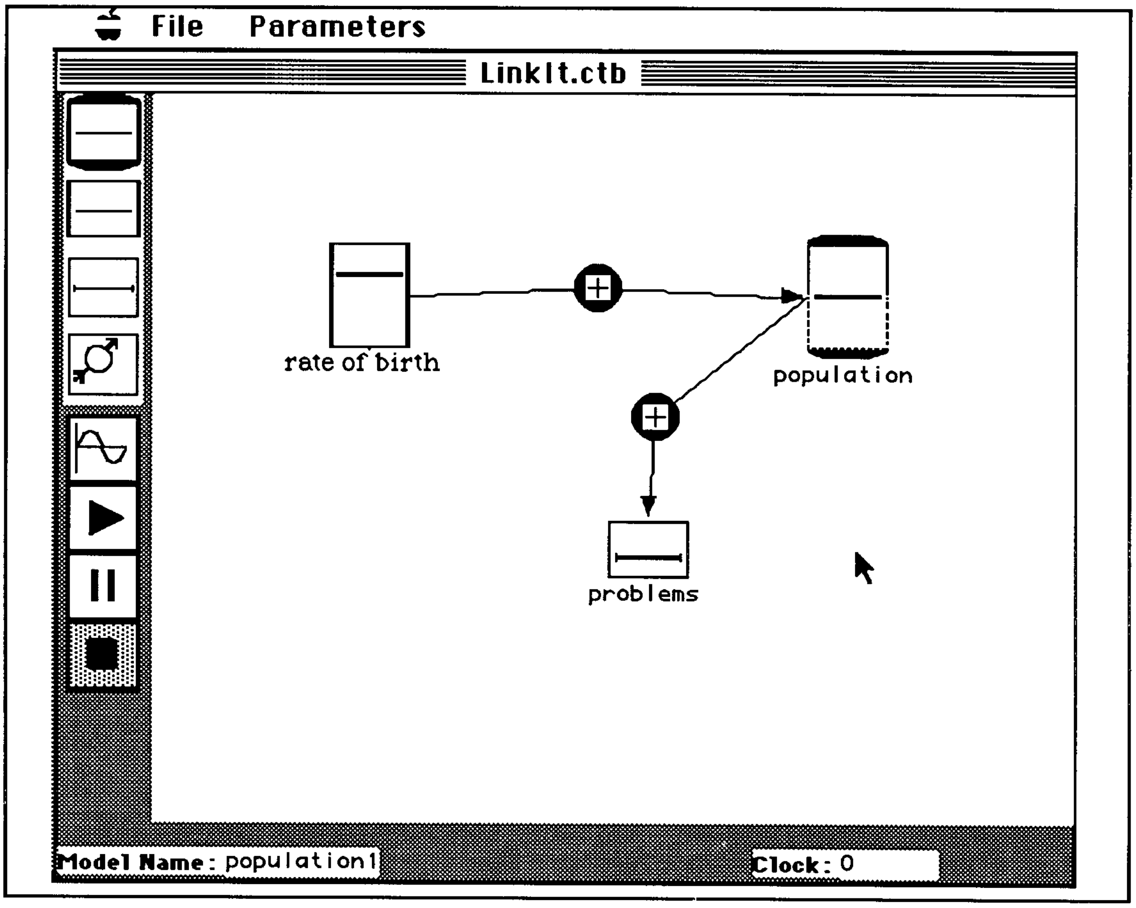


Figure 4.21: An example of a simple model about population before running

The initial internal values (at time = t-1) are:
rate of birth ('immediate-any value' variable; independent) := 0.5;
population ('gradual-bigger than zero' variable, dependent; one ingoing positive link) := 0;
problems ('GONOGO' variable; dependent; one ingoing positive link) := 0. This variable has also a threshold value := 0.285;
weight of the link between *rate of birth* and *population* := + 1 (normal);
weight of the link between *population* and *problems* := + 1 (normal);
Damping of the 'gradual' variable := 0.

<u>Calculating the next internal value of *rate of birth*:</u>

Because it is an independent variable with Damping = 0 then its next value is automatically set to the value of the previous iteraction (t-1).
Rate of birth $_{t=0}$:= 0.5

<u>Calculating the next internal value of *population* :</u>

- It is a gradual variable so the equation to be used is (1):
$\text{population}_{t=0}$:= population_{t-1} + DT ***Input* S - Damping**
- Population has just one positive ingoing link and its causal factor is rate of birth. So its Input is: **Input** := 0.5 * 1
- It is an 'any value' variable. So the squashing function associated to it is:
S := 1 - population_{t-1} = 1 - 0.0 = 1.0
- It does not have a Damping, so **Damping** = 0.
- Substituting the values above in the equation (1):
$\text{population}_{t=0}$:=**0 + 0.1 * 0.5 * 1.0 - 0.0 = 0.05**

<u>Calculating the next internal value of *problems* :</u>

- It is a 'GONOGO' variable so the equation to be used is (3):
$\text{problems}_{t=0}$:= **Input** {if **Input** > Threshold(a_j)}
- *Problems* has just one positive ingoing link and its causal factor is *population.* So its Input is: **Input** := 0.0 * 1 = 0.0
- Because the value of Input is smaller than the value of the threshold of problems, the equation to be used is (3.2): $\text{problems}_{t=0}$:=**nill**

Redrawing the variables on the screen:

The values of *rate of birth* and *population* have to be converted to 'real world' values. To do this the system uses the following algorithm (NewVal is the calculated internal value of the variable; OFFSETZERO is the number of pixels between the upper border of the variable box and its middle: 22):

```
if newVal = 0 then NewOffset := OFFSETRZERO
  else NewOffset := OFFSETRZERO - 21*newVal
endif
```

The NewOffset for *population* is : 22 -21 * 0.05 = 20.95 = 21. So, although the internal value of *population* has changed from 0.0 to 0.05, its real world value is not going to change. In other words, the level of *population* will stay in the same place at the end of the first iteration (21 dots below the upper part of a variable box corresponds exactly to the middle of the box). However, by the next iteration (t=1) the internal value of *population* is going to be 0.1 which makes its NewOffset = 20. So its associated level on the screen will move one line up (one line above the middle of the box).

The NewOffset for *rate of birth* is: 22 - 21 * 0.5 = 11.5 = 12. So, the level of birth will be positioned 12 pixels below the upper part of the box (the same place it used to be).

Problems is a GONOGO variable, so its NewOffset has to be compared with the value of its trigger level in order to see if the box has to be highlighted. In this case (when t=0) it is not going to change.

By the thirteenth iteration *problems* will turn on and after 88 iterations the level of *population* will reach the top of the variable box (its internal value will be equal to 1). At this time the model will have the following appearance (Figure 4.22):

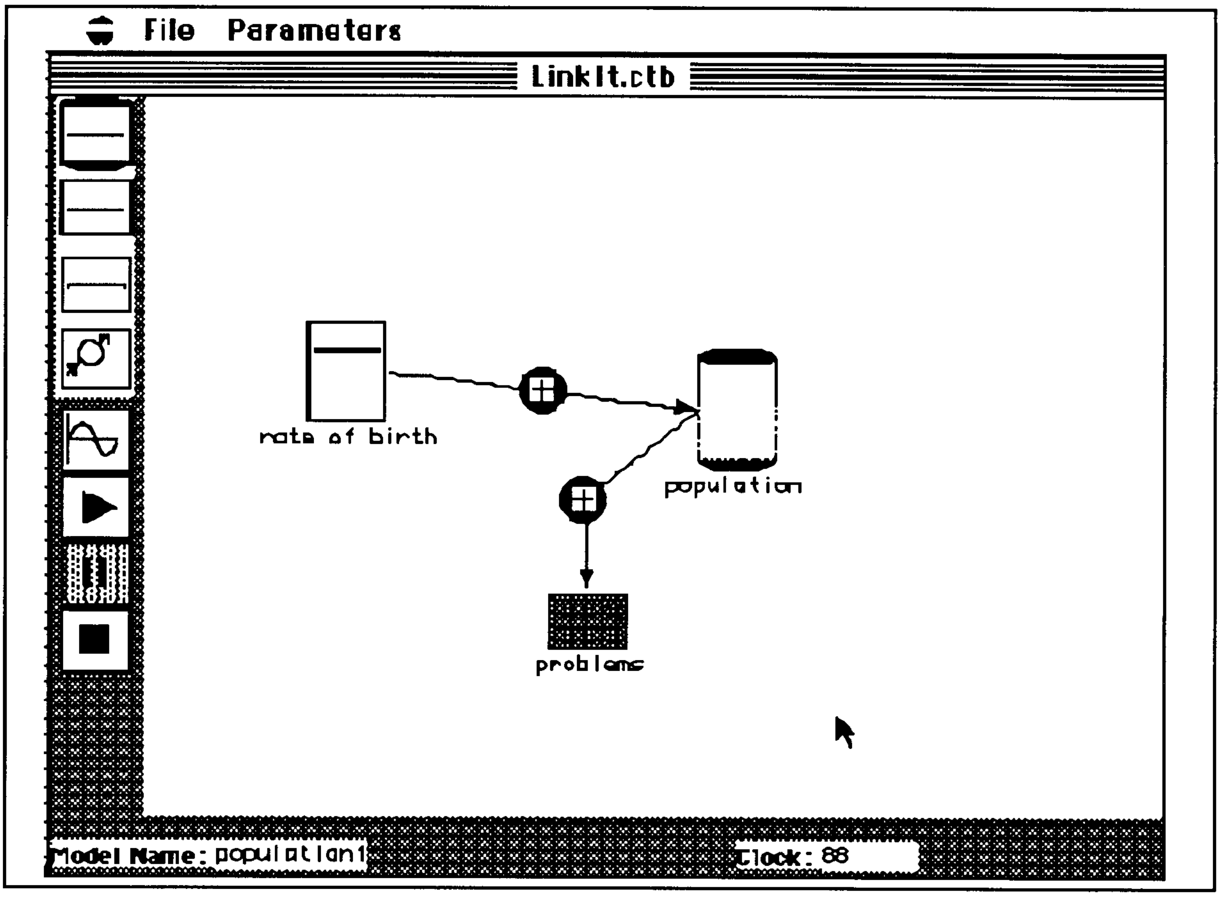


Figure 4.22: The same model about population after 88 iterations

Chapter 5:

PRELIMINARY STUDY AND RESULTS

5.1 INTRODUCTION

The aim of the study described here is mainly to throw some light on aspects of the interface and design of LinkIt I, although some learning issues are also considered.

First the research questions and the description of the study as a whole (methodology) are presented, followed by a synthesis of the analysis of the interviews of the groups involved. For reasons of space, the full details of the analysis are not given here.

5.2 RESEARCH QUESTIONS

The questions addressed in this introductory study concern the aspects of the software that have to do with the interaction between the user and the system and the use of its building blocks to construct/think about models.

The research questions are divided into four different groups. The first group contains questions more related to the use of the LinkIt interface. The second and third groups aim to shed some light on the use of the main components of the system: variables and links. The fourth group is more concerned with the process of criticising and creating models[1] .

Questions About the LinkIt Interface

1) Is it clear to use and easy to master ?

2) Are the control panel buttons meaningful ?

3) Do the different cursor shapes help users to perceive which operation is valid at the moment ?

4) Do the cursor shapes help users to execute the relevant operation ?

5) Are the variable shapes meaningful (in representing their properties) ?

6) Are there new features that could be added to the system in order to help to construct models ?

7) Are there new features that could be added to the system in order to facilitate understanding of the models being constructed ?

8) What errors do users commonly make (1- Initially ; 2 - Habitually) ?

Questions About The Behaviour Of The Variables

1) What are the students' ideas of the behaviour of the variables ? Do these ideas change during the process of modelling a problem ? Do they change after using the system ?

2) Do students understand qualitatively the mathematics that controls the behaviour of the variables ?

3) How do students interpret the zero of a variable ?

4) How do students interpret the input combinations of a variable ? What kind of reasons do they have for changing them ?

5) Why do students choose a certain type of variable ? What kind of reasons do they have for changing it ?

6) Why do students choose a certain range for a variable ? What kind of reasons do they have for changing it ?

---

[1] Although these questions appear here in separate groups, during the analyses they are sometimes addressed together in order to keep them in a relevant context.

Questions About The Behaviour Of Links

1) Do students understand how links work ?

2) How do students interpret the sign of a link ? Does this interpretation change during the process of modelling a problem ? Does this interpretation change after using the system ?

Questions About Modelling

1) Can students interpret LinkIt models ?

2) Can students use LinkIt to think about situations to be modelled ?

5.3 METHODOLOGY

5.3.1 Sample

The Preliminary study was based at the Colégio de Aplicação da U.F.R.J., which is a typical middle class school in Rio de Janeiro, Brazil. Fourteen 13-18 year old students were involved in the experiment.

The students worked in groups of two. They were grouped according to their level in the school. Four groups were in the middle of secondary school (16 to 18 years old) and three groups were approaching the end of primary school (13 to 14 years old)[2].

All students mentioned that they already had some kind of experience using computers. However, during the experiment I noticed that some of them were not very skilled with pointing and controlling the mouse. None of them had used a modelling system before.

5.3.2 Tasks

All groups performed the same tasks, namely: *Introductory Task*, *Exploratory Task* and *Expressive Task*. Below is presented a detailed description of each task, in the form of the guidelines used to conduct the tasks.

5.3.2.1 Task 1: Introductory Task

This was the first task performed by the students. The goal here was to permit them to become familiar with the idea of modelling with LinkIt by exploring its objects, their properties and the main features of the interface. This task was divided into five activities, each focusing on different aspects of the software.

[2] The Brazilian educational system is basically divided into 2 levels. Primary school - 8 years of compulsory study (8 years old to 15 years old) and Secondary school - 3 years (16 years old to 18 years old).

The general approach in this task was to present to the students a sequence of daily-life problems and construct with them causal-loop diagrams to represent these problems.

In the beginning of the work I took control of the system and later (activity 2) the students began to control it.

5.3.2.1.1 Activity 1: Construct a Simple Model about Pollution and Disease

The goal of this activity was to introduce 'gradual' and 'immediate' variables and show how the interface worked.

The approach was to start asking about "the causes of pollution in big cities" and begin to construct a model about this problem:

- What causes pollution in big cities ?
    - Create a model starting with an 'immediate' variable *pollution*
    - Run the model using at each time different values for the variables
    - Show graphs of the variables for different values.
- Do you think pollution works like this in real life?
    - Change the type *pollution* to 'gradual'
    - Run the model using different values for the variables
    - Show graphs of the variables for different values (focusing on the difference between the previous type of *pollution* and its present type).
- What else contributes to pollution in big cities ?
    - Change the model by creating new variable(s)
    - Run the model using different values for the variables
    - Show graphs of the variables for different values .
- What do you think about the input combination for pollution? Do you think it is behaving correctly when we have, for instance, a maximum value for one of its input and a minimum value for the other one ?
    - Run the model with different values for the input variables and discuss the behaviour of *pollution*
    - Change the input combination of *pollution* (from 'average' to 'need all' and 'need any')
    - Run the model again
    - Save the model to use it later.

The features of the system emphasised in this activity were:

- Work with different kinds of variables (GRADUAL and IMMEDIATE) to show that they behave differently
- Show the ideas behind the input combinations of a variable.

The features of the interface emphasised in this activity were:

- Create variables and links
- Modify attributes of variables
- Run a model
- Show a graph of a variable
- Stop a model; Reinitialise a model
- Drag links and variable boxes
- Save a model.

5.3.2.1.2 Activity 2: Explore the Use of 'immediate' and 'gradual' Variables

The goal of this activity was to permit the students to construct simple models in order to perceive, at a deeper level, the behavioural differences between 'immediate' and 'gradual' variables.

The approach to this activity was to present to the students a card containing a drawing of a named variable and ask them to create a model with LinkIt including that variable:

- What do you think X causes ? or What do you think causes Y? (X and Y will be substituted by the appropriate variable name in each case shown below).
  - Let them construct simple models by themselves
  - Run the model with different values for the input variables and discuss the behaviour of the variables
  - Change the input combination of some variables
  - Run the model again
  - Save the models to use them later.

The notation used below to represent relationships in description of models is as follows:

(Same) - Same direction link

(Oppos) - Opposite direction link

(Cum) - Cumulative link

(GTo) - Go together link

Add - Combining links with 'add'[3]

Mult - Combining links with 'multiplication'

Av - Combining links with 'average'

---

[3] "Add", "Mult" and "Av" were used in the core study only.

The model-themes shown to the students were:

- *money* --- **(Same)** ---> ? (a)
- *unemployment* --- **(Same)** ---> ? (b)
- *jobs offered* --- **(Same)** ---> ? (c)
- *use of seatbelts on the roads* --- **(Oppos)** --> ? (d)
- ? --- **(Same)** ---> *life expectation* (e)
- ? --- **(Same)** ---> *rust* (f)
- ? --- **(Same)** ---> *energy bill* (g)

The features of the system emphasised in this activity were:

- Work with different kinds of variables and perceive that they behave differently
- Work with different links
- Work with different input combinations.

The features of the interface emphasised in this activity were:

- Create variables and links
- Modify variables' attributes
- Run a model
- Show graph of a variable
- Stop a model; Reinitialise a model
- Drag links and variable boxes
- Save a model.

5.3.2.1.3 Activity 3: Use of 'Positive' and 'Negative' Links

The goal of this activity was to show the differences between 'positive' and 'negative' links. The approach was to use the contrary ideas behind the examples (b) and (c) above and introduce the 'negative' link:

- Tell them: Let's have a look at the two models we worked with before, the models about *unemployment* and *jobs offering*. Is it true to say that the variables *employment* and *jobs offering* work contrarily ? I mean if one of them causes something to increase, does the other one cause this same thing to decrease ?
  - Let them load the models and reason about them
  - Modify the name of the dependent variable of model *jobs offering* (c) in order to be the same as model (b).
- Now we have two contrary ideas causing the same effect on the same variable. Do you think this is correct ? What do you think we could change in model (c), apart from its name, to make it function correctly ?
  - Change the sign of the link in model (c).

• Now if I change the sign of the link in model (b), what else do I have to change in this model to make it function correctly ?

- Change the sign of the link in model (b)
- Change the name of the dependent variable in model (b)

The feature of the system emphasised in this activity was:

- Perceive the differences between 'positive' and 'negative' links.

The features of the interface emphasised in this activity were:

- Create links
- Modify variables' attributes
- Naming and renaming variables
- Run a model
- Stop a model; Reinitialise a model
- Show graph of a variable
- Save a model.

5.3.2.1.4 Activity 4: Introduce the Idea of "causal-loop feedback"

The goal of this activity was to show the students the idea of causal-loop feedback. The approach was to make further improvements to the model about Pollution in big cities (activity 1).

• Ask them: Is there any consequence for the people living in polluted cities ? I mean if the pollution in a big city increases what happens to the people living there ? (Expect answers related to the health of the population).

- Let them load the model "Pollution"
- Let them choose one kind of variable and introduce it in the model.

• Do you think the kind of variable is correct ? Why ?

- Run the model in order to see if the type of the new variable is correct
- Change its type if necessary.

• When health begins to decrease very much, the papers, magazines and TV begin to report these things, don't they ? What begins to happen to the population ?

- Choose one kind of variable and introduce it in the model.

• Do you think the kind of variable is correct ? Why ?

- Run the model in order to see if the type of the new variable is correct
- Change its type if necessary.

• Now we have people dying because of the pollution and the concern about pollution is getting greater and greater. Do you think it has any effect on the causes of pollution?

- Create a feedback loop
- Tell them what they have created
- Run the model for different values of the variables of the system
- Save the model with a different name.

The features of the system emphasised in this activity were:

- Modify models previously constructed
- Use of causal-loop feedback.

The features of the interface emphasised in this activity were:

- Load a model
- Create new variables
- Create new links (positive and negative)
- Change the range of variables
- Change input combination
- Drag variables and links
- Save a model with a different name.

5.3.2.1.5 Activity 5: Introduce the Use of 'GONOGO' Variables

This activity aimed to introduce the use of 'GONOGO' variables. The approach was to show the students the model about pollution already constructed and start making some improvements to it by discussing what happens when pollution begins to be very high:

- Do you think people get alarmed about the consequences of the pollution on their health as soon as *health* begins to decrease or it is something that begins to happen after a certain level ?
    - Load the model "Pollution and Disease"
    - Tell them about the 'GONOGO' variable and ask them to introduce it in the model as a critical value representing the onset of *over-population*
    - Run the model with different values of the 'GONOGO' variable
    - Show graph of some variables and bring their attention to how the variables behave when the 'GONOGO' variable is active or not
    - Save the model.

The feature of the system emphasised in this activity was:

- Use of 'GONOGO' variables.

The features of the interface emphasised in this activity were:

- Load a model
- Create new variables ('GONOGO')
- Create new links (positive and negative)
- Change the range of variables
- Change input-combination
- Drag variables and links
- Save the model .

5.3.2.2 Task 2: Exploratory Task

This is the first Task to be presented to the students in the second meeting.

The goal of this task was to make the students analyse and criticise a pre-prepared model about the water-cycle (see Figure 5.1). The initial model only contained 'immediate' variables and was presented to the students together with a support text about the theme extracted from one of their text books (see Appendix A - Support text 1):

- After reading the text, ask them whether the model is in accordance with their ideas about the water-cycle.
  - If they say YES, ask them to explain the model and point out some possible problems with its behaviour
  - If they say NO, let them try to make some modifications and keep asking them the reasons for what they are doing.

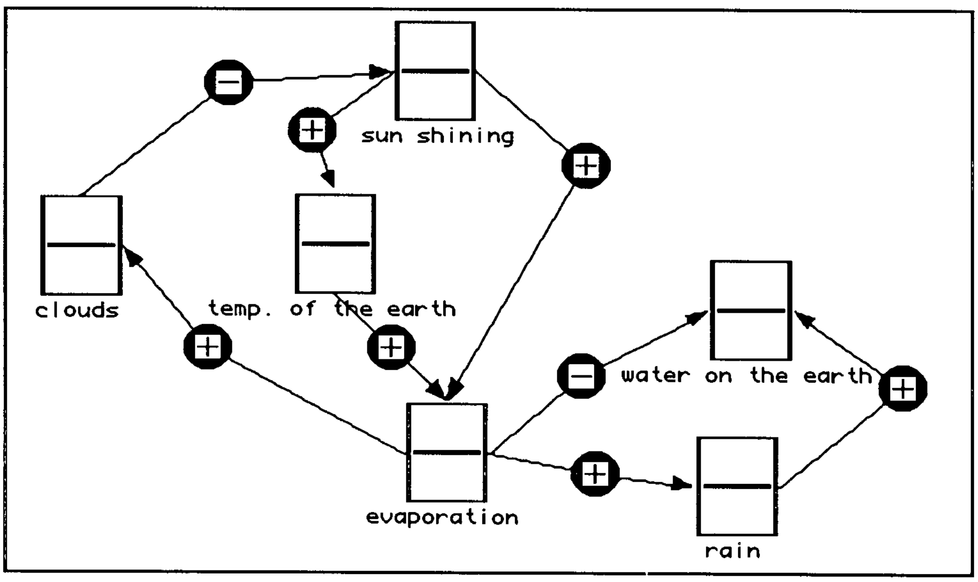


Figure 5.1: Model about water-cycle presented to the students

The feature of the system emphasised in this activity was:

- Work with different kinds and values of variables and links, to make the students perceive that these objects, combined with other functions of the system such as Run and Graph, can help them to reason about a model.

The features of the interface emphasised in this activity were:

- Modify some attributes of variables and links
- Create new variables and links
- Drag objects
- Run a model
- Stop a model; reinitialise a model
- Show graph of a variable
- Save a model.

5.3.2.3 Task 3: Expressive Task

In this task the students were challenged to construct on their own a model about a Nomad population. A support text was given to them. After reading the text, they had to construct a model to represent their ideas about the problem and those existing in the text.

- During the process of construction I asked questions to get them to "think aloud" about what they were doing.

The feature of the system emphasised in this activity was:

- Explore the different possibilities of the system in order to help them to reason about the model being constructed.

The features of the interface emphasised in this activity were:

- Create new variables and links
- Modify some attributes of variables and links
- Drag objects
- Run a model
- Stop a model; Reinitialise a model
- Show graph of a variable
- Save a model.

5.3.3 Procedure and Design

The introductory study was carried out during the months of May, June and beginning of July/1994 at the Colégio de Aplicação da U.F.R.J. in Rio de Janeiro, Brazil. Fourteen students took part. The students were grouped in pairs in order to have a better interaction between them and the computer (Hoyles & Sutherland, 1992). Four groups were aged 15-18 and were in the middle of secondary school. The other 3 groups aged 13-14 years old and were at the end of primary school.

The experiment with each pair of students was carried out in 2 sessions, although there was a case where 3 meetings were necessary. Each session lasted approximately 2 to 2.5 hours with typically 2 meetings per week.

All groups performed the same tasks described in section 5.3.2. In the first session they did Task 1 and in the second session, Tasks 2 and 3.

All students came to all sessions, with the exception of Group G (see Table 5.1), where only one student appeared for the second meeting.

	Name	Age	Sex	Meetings
Group A	Cristiano Gustavo	15 18	M M	2
Group B	Ivan Marcelo	17 16	M M	2
Group C	Maria Liza	16 16	F F	2
Group D	Gabriel Renata	16 17	M F	2
Group E	Giselle Laura	13 13	F F	3
Group F	Iuri Daniel	14 13	M M	2
Group G	Renata Fernanda	13 13	F F	2

Table 5.1: Description of the groups and number of meetings needed

At the beginning of the first session I explained that we were going to use a computer modelling system developed by me and the idea of the experiment was not to judge whether they were right or wrong about the models they were constructing, but rather I was interested in what kinds of thoughts and reasoning they used during their interaction with the software. In order to do so I asked them to talk aloud (Gomoll, 1990) as much as possible about what they were thinking.

The tasks were performed in the same order as they appear in Section 5.3.2. The description presented there served as a guideline to conduct the interviews. Thus there were moments when some features of the system had to be introduced earlier than was planned or were not introduced at all (e.g. the graph function was not used by the younger students - 13 to 14 years old).

All meetings were tape-recorded and some groups were also video-recorded when executing Tasks 2 and 3. During the sessions we kept saving the models that were being constructed. This information was used during the analysis to have a better understanding of what the students were talking about and how their ideas about a certain problem evolved.

5.4 ANALYSIS AND RESULTS

The main purpose of this analysis is to see how the students understood the objects of the system and how they used them to construct and/ or modify models (see also Section 3.7). It is expected that this analysis will give feedback for further consideration of the necessary building blocks that a semi-quantitative modelling system has to possess in order to be used with a wide variety of educationally interesting problems.

For the sake of brevity, what is presented here is a synthesis based on the detailed analyses of five groups interviewed. Appendix C presents the analysis of one of the groups involved in this study. The five analyses from which this synthesis was made can be found in (Sampaio, 1995).

The subsections presented below are in accordance with the four groups of research questions shown in 5.2.

The groups in each passage are identified by capital letters (from "A" to "G" - see Table 5.1). The subjects of the respective groups are identified by the first letter of his/her name. The interviewer is identified by the letter "F"

5.4.1 Using the System Interface

In a general way it can be said that the students managed very well to control and use the interface to perform the tasks. From the beginning they took over the system and tried things by themselves. The interventions made were mainly to show them new features of the system.

Inconsistency of the interface

There were some repeated actions among the groups that could be related to inconsistency of the interface and/or its implementation. One of these inconsistencies concerns the different ways the system provides for performing similar tasks. Whenever the user creates a new variable, LinkIt presents a Dialogue box asking the name of that variable. After typing it the user has to press the return key. However, when a name of a LinkIt file is requested, the user has to press an OK button (after typing the name) in order for the system to accept the new name. What happened was that some students did not notice this difference and normally pressed the return key and kept waiting for the system to complete the operation. An example of this situation happened with Group A when constructing a model during Task 1 activity 2:

G: " *Money* (name of the model)... *Is it saved or not ?* "

C: " *I think you have to press* (the OK button)"

G: " *Yes. You have to press OK*"

C: " *It is different* (here)*isn't it ?*"

G: " *Yes , it is*"

It seems here that an important rule about consistency in the design of the system interface is being violated: "similar tasks have to be performed in a similar way" (Shneiderman, 1992; Tognazzini, 1990).

<u>System time response</u>

The system's time response to perform some operations or to judge a certain action taken by the user, also caused some problems. It becomes more critical when the user tries to move an object on the screen or to change some of its properties. In these cases the system has to calculate and redraw - in the same order in which they were created - all objects that are overlapped by the object moved. The amount of time taken by the system to perform this operation was big enough to bother the students, making them remark unfavourably on it and to try to find ways of getting around it. An extreme case occurred with Group A when they decided to reconstruct an entire model about the nomad population problem (Task 3) just so that they did not have to wait for the system to redraw some boxes they wanted to move and rename.

After creating almost all variables they decided to delete the model and create everything again because they didn't like the way it was organised on the screen.

F: " *Why don't you move the objects on the screen ?*"

G: " *Oh, no! It is too much work*".

There are two possible computational causes for this problem. The first is that the algorithm used to calculate the objects that need to be redrawn is the same regardless of the object the user is working with. The second is that the machine used during the experiment was a Macintosh LC II! with only 2 megabytes of RAM memory with some processes running in the background.

Control panel and its buttons

Sometimes the students did not notice that a certain button was already selected and went to the control panel and tried to select it again. One possible reason for this behaviour is that the way the system highlights a selected button is not "clear" enough to make the students notice it. Actually the pattern used to highlight buttons is quite similar to the background used in the control panel (In Figure 2.2 the second button from above is selected). However the cT environment does not give much choice of patterns to use. Either they are very light or very dark.

Another aspect related to the control panel is the different meanings the students gave to the execution buttons provided. The 'pause' button was interpreted by one group (Group C, during Task 3) as a 'stop' button. They were in doubt whether they had to press it again, or to press the 'run' button in order to make the system resume the iteration of the model:

> M: " *Pause here and increase rain ?*"
>
> L: " *Yes, isn't it ?* (...) *is OK here ?*" (moving the 'level')
>
> M: " *Yes, it is OK there.*"
>
> L: " *So what do I do now ?* (...)"
>
> M: " *Press pause again. No,* (press) *play*"
>
> L: " *To leave it from the pause* (situation), *press the pause again or press the play ?*"

At that time they had to press the 'run' button. In consequence, following that interview the software was changed in order to make the 'pause' button behave as a toggle where the first press pauses the iteration and the next one makes it resume[4].

Not all students noticed the set of operations related to the 'reset' button. Actually, there was a special case where the group mentioned that it only reset the system clock. A partial explanation of this problem can be the (likely) experience of the students with other appliances (e.g. VCRs, CD players) where a similar button is used solely to stop a certain function such as playing a tape and not something that resets the operation that has been performed (such as stop playing and rewind to the beginning of the tape or the music).

Cursor shape

It seems that the different cursor appearance sometimes hindered the users in performing their tasks. Because of the way cT works with cursors, the students had some problems in performing some operations such as moving the

---

[4] The user can also press the 'run' button after pressing the 'pause' button to make the system continue to iterate

'level indicator'. According to the cT implementation, the "hot region" of a cursor is its upper part. This does not cause any problem when the cursor has an arrow shape, because the user intuitively uses it to point to objects to perform a desired operation. However when the cursor assumes other shapes such as a box (resembling the different variable boxes of the system) or a Cartesian graph (when the graph function is ON), the intuitive pointing region does not exist any more. At this time the students tried to use other parts of the cursor to point to objects, but because its upper part remained its "hot region", the system (in most cases) understood differently what the user was trying to do and performed a different operation. This situation was specially awkward for the students when they tried to move a level of a box and the system understood it as moving the whole box. Not only did the system not do what they wanted but they also had to wait for the system to redraw all overlapped objects.

<u>New features</u>

During the interviews the need for a feature that could permit the students to follow the evolution of the model over a certain period of time became clear. This idea can be very helpful especially when there is a model with many variables and users want to pay attention to different groups of them at different times. Related to this idea is the need to give the user the possibility to reuse the initial conditions of a certain simulation instead of having to reset the model and set all variables again. A possible way to implement this feature is by recording the initial conditions of the last simulation and let the user click on a certain button on the control panel whenever he/she wants to restore it.

5.4.2 Variables and Their Properties

<u>'Gradual' and 'immediate' variables</u>

It seems that the students understood the fundamental characteristic of 'gradual' and 'immediate' variables in a very qualitative way. For them, 'gradual' variables were used to represent amounts or events that varied little by little as time went on, seeing them as gradual in time rather than cumulative, whilst 'immediate' variables served to represent things that move fast and/or in big steps. Group E, during Task 1, talked about these two kinds of variables in this way:

F: " (..) *Do you know this word gradual, is it possible for you to understand what is something gradual ?"*

G: " (It) *Changes little by little "*

F: *" Changes little by little ? and something immediate ?"*

L: *" It is quick. It changes suddenly"*

These ideas fitted in well with most of the tasks the students were working with. However, there were moments where they could not " fix" something in their model just because they did not have a complete understanding of these different types of variables. When this happened, their usual approach was to try different possibilities for the variables and to use the one that suited better to "fix" the problem. These changes, in some situations, forced them to construct another meaning for the nature of the variables already created.

The ideas of accumulating things and a dependent variable following its cause, though very rare, were also employed by the students to justify the type of a certain variable. In the following two passages Group D talked about 'gradual' and 'immediate' variables using the ideas of "accumulating" and "following a cause"[5].

During Task 1 activity 1

F: " (...) *pollution, how it functions in real life ? Is it something that goes...it changes instantaneously or it is something that grows little by little ?"*

G: *" It goes little by little"*

R: *" Yes, because it is cumulative isn't it? So every time more..."*

During Task 1 activity 2

F: *" In this case here, how do you... the type of the variable hunger there, the type of unemployment ? How does it evolve as time goes on ?"*

G: *" It is proportional"*

F: *" What does it mean, "proportional" ? "*

R: *"As much you have unemployment as much you have hungry"* (the number of unemployed people is the number of hungry people)

G: *" Yes. If X people are unemployed then X people are hungry"*

After changing the type of *hunger* to 'immediate' and running the model I tried to confirm their ideas...

F: *"OK. So unemployment and hunger. I mean, for you hunger is something..."*

R: *" It is the direct reflection* (consequence) *of unemployment."*

The students tended to look at a variable in a very local way: If a variable was not behaving in a proper way, the problem, for them, was inside the variable

---

[5] Note that Group D is a pair of older students and perhaps more sophisticated mathematically.

and not, for instance, in the relationship between the variable and its cause(s). An example of this case happened with Group D during Task 2. They changed the variable *temperature* to 'gradual' and after made it 'immediate' again because they did not achieve an expected behaviour with the model:

G: *"The temperature should be gradual..."*
R: *"But isn't it* (gradual) *?"*
G: *"No. It is immediate"*
F: *" Temperature of the earth is now gradual"*

After running the model and not being satisfied with its behaviour...
G: *" (...) this temperature of the earth is gradual. I will change it again to immediate"*
R: *" Yes it didn't make too much difference"*
G: *" Yes. It didn't make any difference"*

A possible way to try to solve this problem is to use typed links instead of typed variables. It seems that with this approach users have to be more explicit about how variables affect one another and also do not have to find other meanings for the variables already created in order to adapt them to the problem they want to solve.

<u>'GONOGO' Variables</u>

'GONOGO' variables were used on many appropriate occasions although some students had problems in understanding the meaning of its level. They had a tendency to interpret the level as "how much of something someone has" and not "how much of something is necessary in order that something that follows happens". An example of this case is shown in the passage below (Group B during Task 1 activity 5). After the explanation Marcelo was still seeing the level representing "how much anger there exists":

F: (explaining the idea of the level of a GONOGO variable) : *"... you have to say what is the level of anger of population* (GONOGO variable) *from where things will begin to happen . Here if you leave it (the level of anger of population) where it is* (zero position), *you are saying that the anger of population is zero "*
M: *"There was nobody angry "*

It seems that in these situations they were again trying to use the consistency rule: There is a strong resemblance in the appearance of the three variables (but not in their behaviour) and therefore the students were trying to transfer their knowledge about the meaning of the level of 'gradual' and 'immediate' variables (which were introduced first) to the new one.

The uses that can be made of 'GONOGO' variables are still very limited. Some students, for instance, suggested that there should be a way to make the *system clock* control its behaviour (when it has to be "on" or "off") and/or to attach different values to its output when it is "on" and "off".

Interpreting the Level

The idea of "zero" of a variable was more naturally associated with the very bottom of a box instead of its middle. The difficulties of understanding the different input combinations for a specific variable also contributed to make it more difficult to find out where was the zero position.

The meaning given to the "zero" point of a variable seemed to vary with what was intended to be represented. In some cases they interpreted it as a normal value or the absence of a certain factor. An example happened during Task 2 (model about water-cycle) where the variable *temperature* in the middle had the meaning of being neither hot nor cold (it was normal) but *clouds in the sky,* with the level in the same position, meant no clouds at all.

Half boxes and level below the middle

The students understood the idea of half boxes and employed 'bigger than zero' in some models, though they never used negative numbers and 'smaller than zero' variables.

Although it seemed not very intuitive, the students could understand that when the level indicator of an 'any value' variable was below the middle it began to pass a negative value to its dependent variable(s). Sometimes they kept a variable as 'any value' - even when it did not make sense to have "below zero" values - just because they wanted to use its negative output to influence the behaviour of another variable in their model.

Input Combination

Some students found it difficult to understand the idea behind 'need all' and 'need any'. Sometimes they interchanged their ideas or simply did not understand them. An example happened with Group D when this was introduced to them:

F: *" How do you think they are calculating here the input combination ?"*

G: (the information box is opened): *" Oh, yes. They're calculating the average, like I said"*

F: " (...) *It is doing the average. There exist two other options: need all or need any. In this situation here. What do you think would be a better input combination than average?"*

R: *" All factors* (causes). *Chemicals as well as pollution* (smoke) *of cars* (are needed)"

F: *" It would be necessary all or need any ?*

G: *" Need any"*

R: *" I am not understanding it* (...) *what would be input combination anyway ? What would it be ?"*

The problems about understanding the input combination can be related to the abstractness of the ideas of "need all" and "need any" and the fact that the translation of the 'need any' option into Portuguese had much more a meaning of "need just one". During the work with the system some students preferred to talk about the input combination using names common to them in their mathematics classes such as "add" , "subtract" and "average".

Rate of change

To develop a scientific concept about rate of change is not an easy task. To start talking about it in a qualitative way can be a good starting point. Although the students did not talk about rate of change very much in this study, it was observed that they sometimes did do so by making comments about the variables and their levels in at least two different ways:

• When they talked about the different speed of the level of a variable for different values of its input;

• When they talked about things accumulating.

The two passages below are examples of these two cases:

Group D was constructing a model about money (Task 1-activity 2). *Money* and *wealthy* were connected by a positive feedback loop (see Figure 5.2).

F: *" So, you connected money to wealthy and wealthy to money ?"*

G: *" When we changed the level* (of money), *we, what we put here in the level was the factor of increasing and not the quantity of money. If you put a little here* (set the level of money), *it will be how much it will increase. The proportion that it will increase. It will increase just a little. If I put it in the maximum, it will increase very fast."*

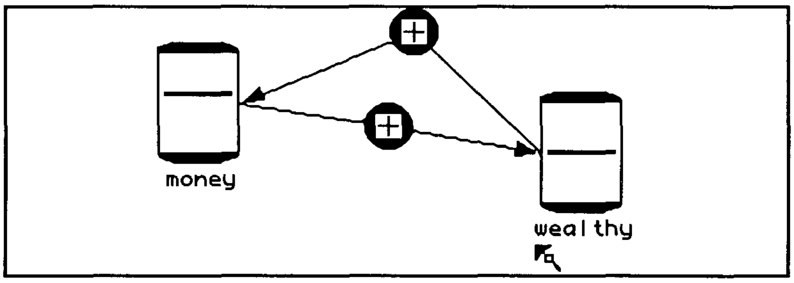


Figure 5.2: Group D: Model about money with a feedback loop

The next passage is from Group A, working with the 'pollution' model in Task 4.

It was suggested to introduce a consequence of pollution and the students decided to introduce a variable called *lung diseases*. When asked about the type of the variable they said:

G: " *Gradual, disease ? Yes* (it is) *gradual. It is not immediate because we keep on breathing, breathing and with the time we are becoming ill .*"

...

C: " *Yeah. Exactly. It's something cumulative.*"

However there was also an extreme behaviour in the other direction. One student could not understand why a dependent variable was varying while its cause was constant.

5.4.3 Links and Their Properties

The students did not have problems in understanding that links were to create a causal relationship between variables. However some of them, in the beginning of the work, got confused about the direction of the relation and whether the links went both ways.

The idea behind 'positive' links was easier to understand than 'negative' links. There were cases where the students associated 'negative' links with "something that decreases the dependent variable". An example happened with Group E during Task 1 activity 4:

F: " (...) *why do you think the sign is minus here* (between health and industries) ?"

L: *"Because the worse the health the less the industries. People are going to protest because many people are dying..."*

After creating a new variable called *population urging action*, dependent on *health of the population* (positive link) and connecting it to *industries* and *vehicles* with negative links...

F: *" So, worse the health is the more the population urges action ?"*
L: *"Yes"*
F: *"So the sign here is plus, between health of the population and population urging action ?"*
Both: " Yes"
(...)
L: *" We did this arrow here from population urging action to vehicles"*
F: *" Mm, OK"*
L: *" And we put minus"*
F: *" So, you mean, the more the population urges action the less is the number of vehicles. Is that what you are saying ?"*
Both: " Yes"
G: *"The same as industry"* (link between *population urging action* and *industry*)

(It is important to note that from a mathematical point of view, a minus sign ("-") on a link multiplies the value given to it by -1. A plus sign ("+") multiplies the same value by + 1.)

One reason for this behaviour could be the fact that they were employing their mathematical knowledge where 'plus sign' means "to add or increase" and 'minus sign' means "to subtract or diminish".

The students could understand the idea of the strength of a link and used it in different moments. During the work with the Nomad population problem (Task 3) it was noted that it would be interesting to have at least 3 different values for the strength of the links in order to make clearer the influence of different causal variables.

5.4.4 <u>Modelling</u>

<u>Strategies Employed</u>

The students were very concerned about making their models work (run) in a "correct" way and to achieve this they did not bother very much about the types and properties of variables and links. An example of this behaviour is a situation where they began using a variable as 'bigger than zero' justifying it in terms of "making sense" in the real world, but later if they noticed that the model was not behaving in the way they wanted, they would go and change the range of the variable to 'any value'.

The process of "fixing" a model consisted basically of using a trial and error strategy, making changes to the properties of variables and links until the desired behaviour was achieved. Only when they had their model behaving in the way they wanted, did they try to justify the objects and properties they employed.

Model structure and complexity

The structure and complexity of their models varied very much with the activity they were engaged in. During Activity 2 where the idea was to construct simple models in order to have a better understanding about the behaviour of variables and links, the students only employed 2 or 3 variables in their models. In the last task, when they had more freedom to test their ideas, they constructed quite complex models, mixing different types of variables and links.

However during Task 2, where they had to criticise a model about the water-cycle problem, they could make comments about its behaviour but were not very keen to make changes to it.

Concrete examples and potential formalisation

Their experience of the real world was used in many situations to justify, explain or excuse the behaviour of the models they were working with.

Also some students were able to analyse and think about the variables and links of their models, recognising that they were only working with a possible representation of reality and that the system is just a tool to represent these ideas (Group B during Task 1 activity 2):

> M: " *Yeah, but here we have already seen that.. the middle* (of a variable box) *is ... is zero, isn't it ? If we put everything to the bottom* (the 'level' of industries), *it is because industry is... would be a variable .... because it* (the system) *doesn't understand that we are talking about industries, it understands that we are working with a variable, and if the variable is below zero then it is cancelling the smoke from cars. It would be like... in this case it would be an action against the industries, like they were taking out the pollution that comes from industries* "

5.5 FINAL REMARKS

In general it can be said that the students were able to learn and control the tool to perform the tasks presented to them.

However there are some aspects of the interface that need to be reconsidered in order to make it more consistent and to bring the system closer to the students' way of thinking. These aspects can be summarised by the following proposals:

- Pursue a more consistent way to interact with the different dialogue boxes;
- Optimise the redrawing algorithms;
- Look for a better metaphor for the buttons of the control panel (specially the ones related to the simulation of a model);
- Think about the idea of typed links;
- Think about the sign of a link to avoid misunderstanding;
- GONOGO variables: Look for another way to represent its value and its threshold. Think about representing "under flow events";
- Look for a better way of representing the "zero" of a variable;
- Look for other ways of describing the input combinations.

Chapter 6 presents the design and implementation of a second version of LinkIt taking into consideration the issues presented above and also including new features that were felt necessary for the use of such an environment in classroom activities.

PART III

DESIGN AND TESTING OF LINKIT II

Part III presents and discusses the design and implementation of a second version of LinkIt (LinkIt II) and a Core study carried out in Brazil with students using this new version of the software.

It is divided into five chapters: Chapter 6 discusses the design and implementation of LinkIt II; Chapter 7 presents the research questions and the methodology of analysis of the Core study; Chapter 8 introduces the tasks used during the Core study; Chapter 9 presents an analysis of the students' understanding, use and manipulation of the software; Chapter 10 continues this analysis but now focusing on students' thinking and learning with LinkIt.

Chapter 6:

LINKIT II: DESIGN AND IMPLEMENTATION

6.1 INTRODUCTION

Prototype II was developed from the analysis of the Preliminary study made with Prototype I. Although much of the desktop and user-system interaction did not change, there were fundamental modifications in the building blocks and their properties.

It is important to note that the same assumptions made in chapters 3 and 4 about the educational tenets for the design of the system are still valid here.

The subset of system dynamics modelling possible to be represented with LinkIt II is the same in comparison with prototype I (see Section 4.4 and 4.5.3.3), however here they can be represented with more accuracy.

The changes introduced in the interface and in the mathematics of the system were mainly to make the system more user-friendly and consistent.

In this chapter I use the same approach as in Chapter 4 where I described the main components of the system and how to operate them together with the design decisions that it was necessary to make.

6.2 PROTOTYPE II - DESCRIPTION OF ITS COMPONENTS

In this section I describe the main components of the system beginning with the existing building blocks. I then consider the interface and how to operate it.

6.2.1 Description of the Objects of the System

LinkIt II has two main system-objects: variable and link. Although they still have the same purposes as in Prototype I, some of their properties were completely modified and new ones were added (see Chart 6.1).

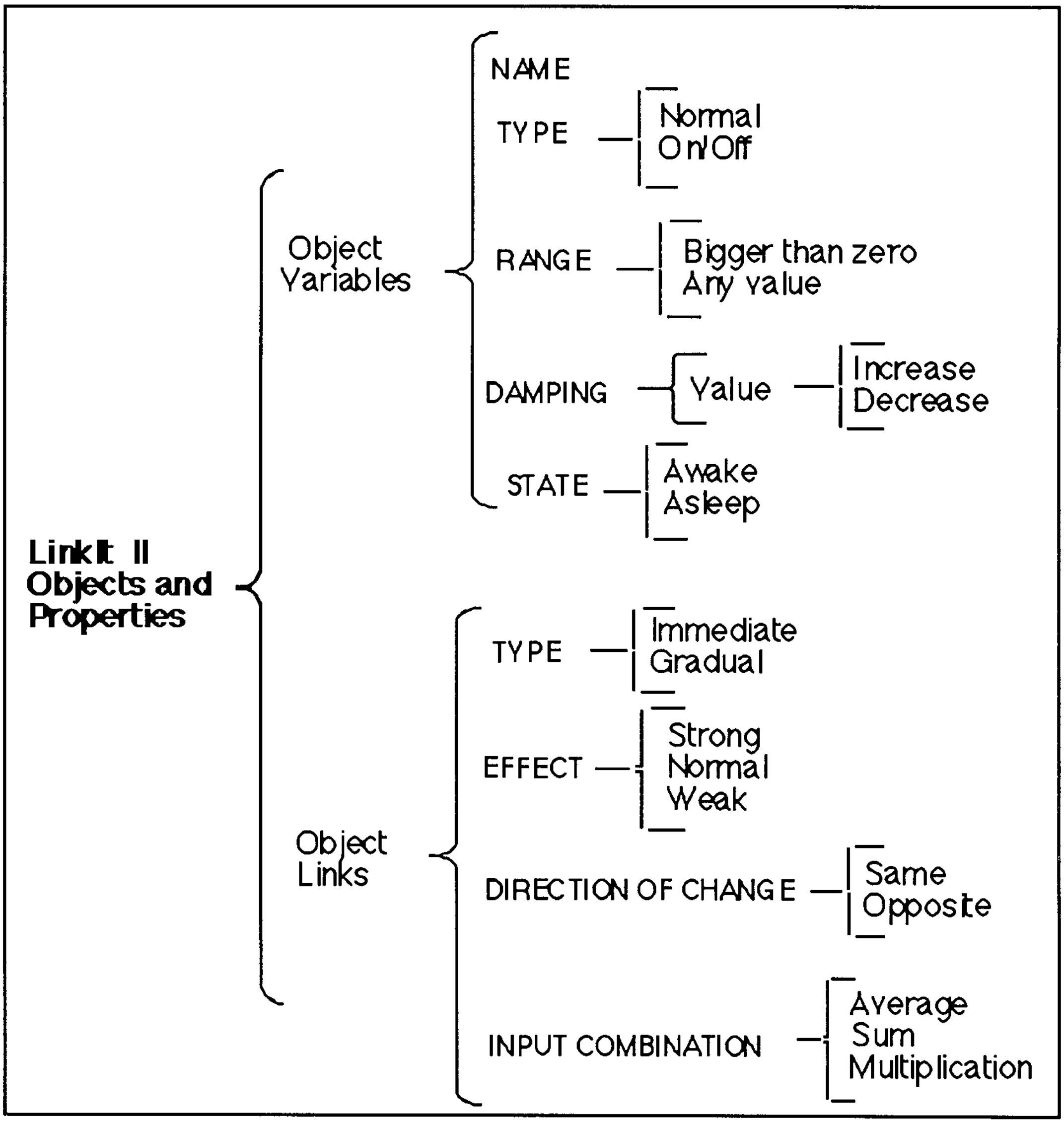
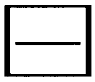


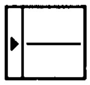
Chart 6.1: Objects and their Properties

In the next subsections I shall describe these two system-objects thoroughly.

6.2.1.1 Object-Variables and Their Properties

LInkIt II has two different types of system-object variables: Smooth and On/Off variables. They are presented to the user as a box (called variable-box) containing one or two level bars inside it.

 Smooth variables can be used to represent any factor about a problem someone wants to model. They are shown on the screen as a rectangular box with a horizontal amount level indicator in the middle.

 On/Off variables can be seen as a special case of Smooth variables. They serve to represent conditional factors that can control the behaviour of dependent variables. Their box is divided into two parts. The left part contains a threshold level indicator (a small triangle) used to define when the variable will trigger and the other part of the box contains the amount level indicator. Both of them can be moved by the user independently .

This variable is visually quite different from the 'GONOGO' variable in LinkIt I (see variable *preoccupation/alarm* in Figure 6.18 and Figures 6.1.c and 6.1.d). Now the amount level is explicitly shown with the variable and can be compared with the threshold level indicator.

When a variable is created its default range is set to 'positive only' values. If the user changes it to 'any value' (through the Information box of the variable), the height of the variable box doubles (see Figure 6.1) and all links that arrive or depart from it are redrawn to fit the new size of the box.

The resting level (internal value = 0) of an 'any value' variable is indicated by two small dots located on the left and right side of the box (see Figure 6.1.b and 6.1.d)

When a variable is created, the system automatically gives it a "*noname* " name. If the user wants to change it he/she has to perform a change name operation (see Section 6.2.4.1). This modification was introduced to give more freedom to the user when creating a model. Now he/she does not need to be "precise" about what they want to be represented by the variable at the very beginning of the process of creation of a model.

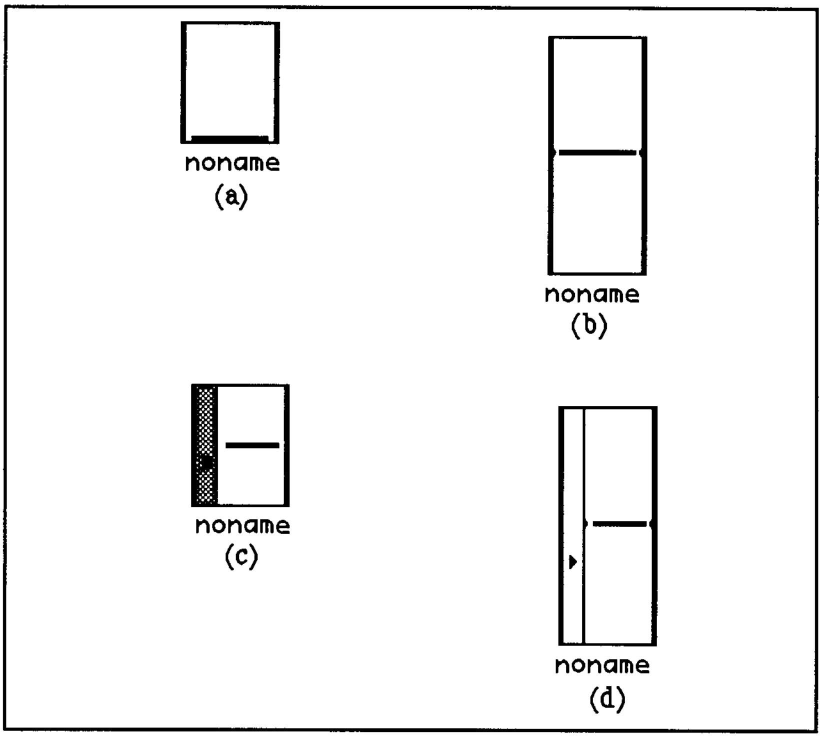


Figure 6.1: (a) Smooth Variable (positive values) (b) Smooth Variable (any value) (c) On/Off Variable (positive values, triggered) (d) On/Off Variable (any value, not triggered)

The properties of an object -variable are (see Figures 6.2, 6.3):
- *Type* - Defines the type of an object-variable. It can be Smooth or On/Off. When a variable is On/Off, a specific set of properties that deals with the triggering event, is automatically added to the Information box of the variable. These properties are: (a) *When to trigger* - Used to define when the variable has to turn on. It can turn on when the amount level indicator is *above* the threshold level indicator or it can turn on when the amount level indicator is *below* the threshold level; and (b) *Output Value* - Defines what will be the output of the variable when it becomes on. It can be *high* (a big value), *low* (small positive value) or *equal to* the amount level indicator (the default output value);
- *Range* - Defines the maximum and minimum values for a specific variable. It can be *bigger than zero* - the internal value of the variable varies in the interval [0, 1] or *any value* - the internal value of the variable varies in the interval [-1, 1]. The default value is 'bigger than zero';
- *Change by itself*- This property replaces the Damping property used in LInkIt I (see "Menu Parameters" in Section 4.5.3). It attaches a natural decay (decrease option) or growth (increase option) to a certain variable, one at a time. The intensity of changing (its value) can be regulated by the user through a slide bar. The default value is zero (no change);

- *State* - Defines whether a variable should be considered in the calculation when the model is running. When a variable is *asleep* it is not considered in the calculation of the next iteration, and the opposite if it is *awake*. When a variable is *asleep* its appearance on the screen changes to a dashed pattern.

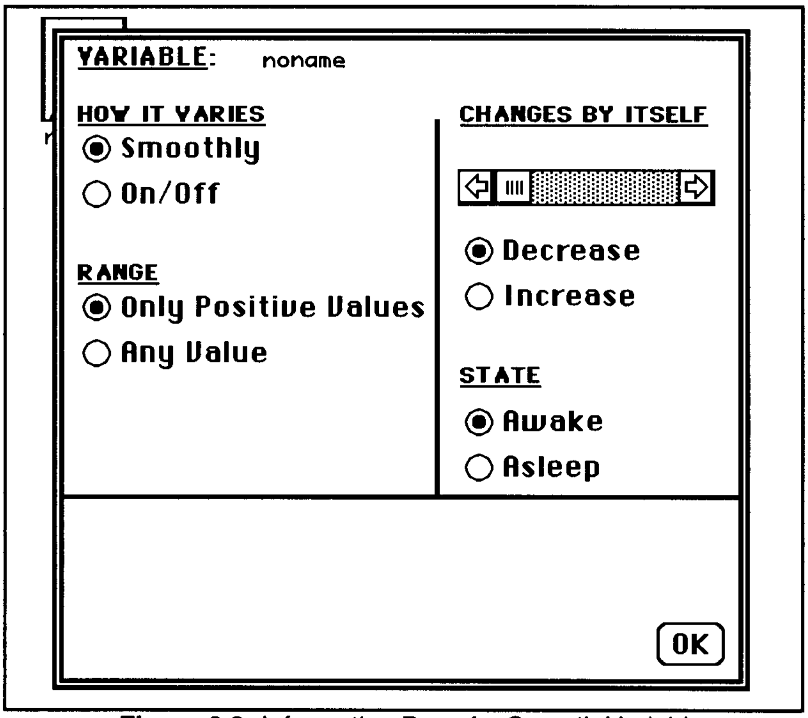


Figure 6.2: Information Box of a Smooth Variable

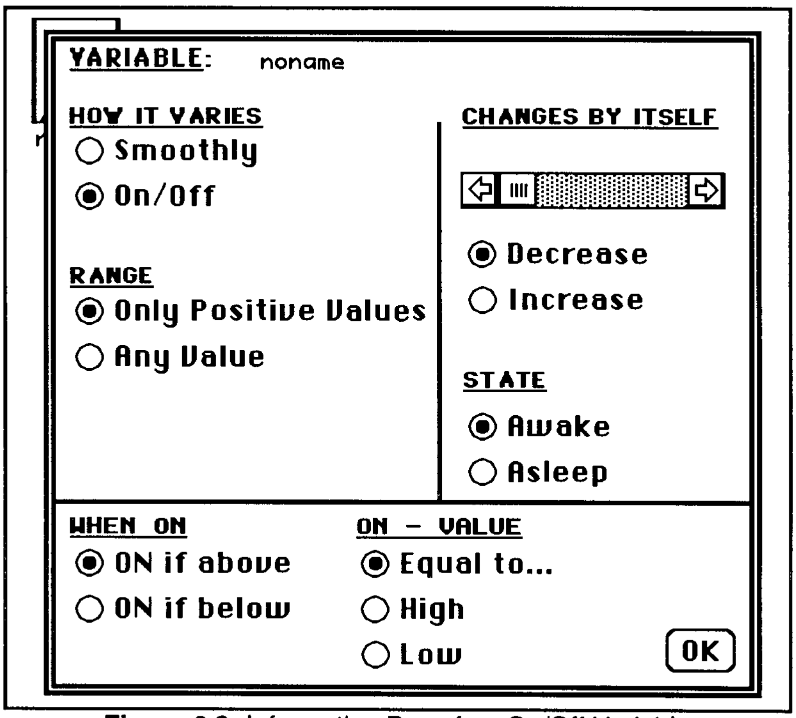


Figure 6.3: Information Box of an On/Off Variable

6.2.1.2 Object-Links and Their Properties

Like Prototype I, links are used to connect variables and define a causal relationship between them. In this version some of their properties were changed and new ones were added. Below there is a description of the properties of a link and a discussion about them.

- *Type* - The system provides two different types of links to represent the ideas of a 'go together' relationship and a 'cumulative' relationship. This is a major change introduced in this version mainly to induce students to be more explicit about how variables affect one another (see Section 5.4.2)[1]. These two links are shown on the screen with a different appearance. 'Go together' links have a square box in the middle and 'cumulative' links use a circle (see the links departing from *pollution* and *car-smoke* in Figure 6.18). 'Go together' links are default but whenever the system detects that a variable already has an incoming link of a certain type, all other new incoming links will be created with the same type as the one which is already there. Similarly, if the type of one link is changed all other links that arrive at the same variable have their type changed;
- *Direction* - This property is used to define how the direction of the dependent variable will be in respect to its cause. It can be of two kinds. The *same direction relationship* is used when the user wants the dependent variable to follow the result of its input (e.g. if the input variable is positive it will pass a positive value for the dependent variable). The *opposite direction relationship* is used when the user wants the dependent variable to follow the opposite (contrary) value of the result of its input (e.g. if the input variable is positive it will pass a negative value for the dependent variable). In the model in Figure 6.18 the relationship between *pollution* and *diseases* can be "read" as: "If *pollution* is high then *diseases* will be high" and the relationship between *diseases* and *health* can be "read" as: " If *diseases* is high then *health* will be low". Whenever the system detects that a 'bigger than zero' variable can assume negative values it prompts a message to the user advising him to change its range to 'any value';
- *Effect* - This property is used to define the strength of a relationship. It has three possibilities: The *normal strength* indicates that the weight of a certain link on the calculation of a dependent variable is equal to 1. The *strong strength* has a weight equal to 2 and the *weak strength* which has a weight equal to 0.5. The size of the arrows or the number of triangles are used to represent the idea of different effects.

---

[1] Instead of having 'immediate' and 'gradual' variables, LinkIt II provides 'go together' and 'cumulative' links.

The direction and strength of a link are visually represented by arrows ('go together' links) or small triangles ('cumulative' links) that go inside the icons of links (see Figures 6.4 and 6.5).

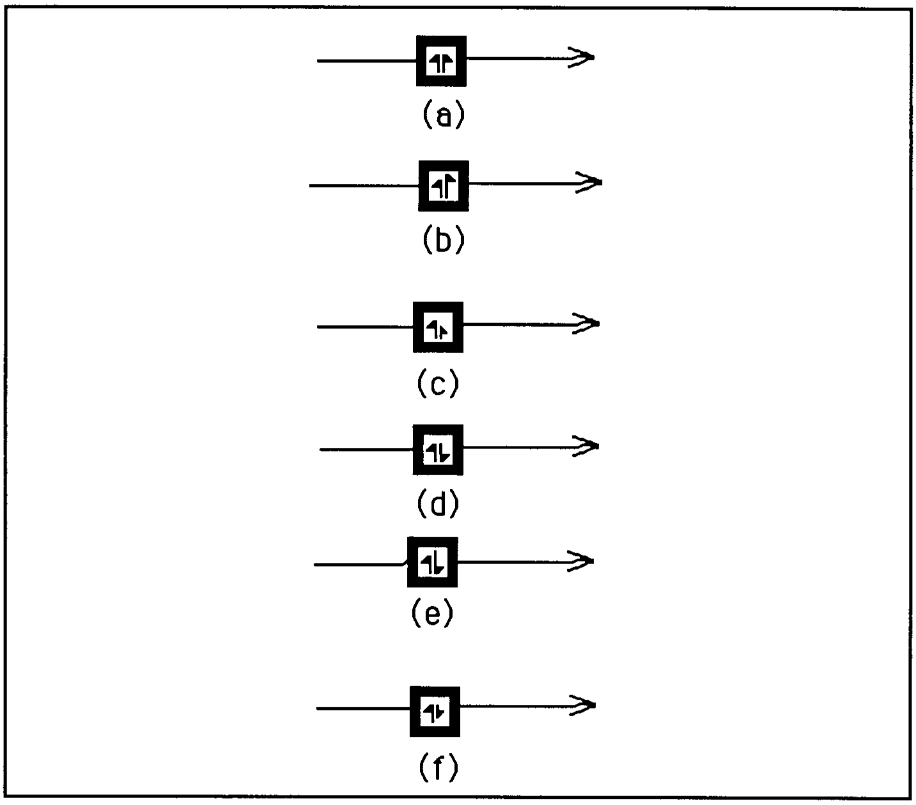


Figure 6.4: Different strengths and directions of cause of 'go together' Links: (a) same direction/ normal strength; (b) same direction/strong strength; (c) same direction/weak strength; (d) opposite direction/ normal strength; (e)opposite direction/strong strength; (f)opposite direction/weak strength

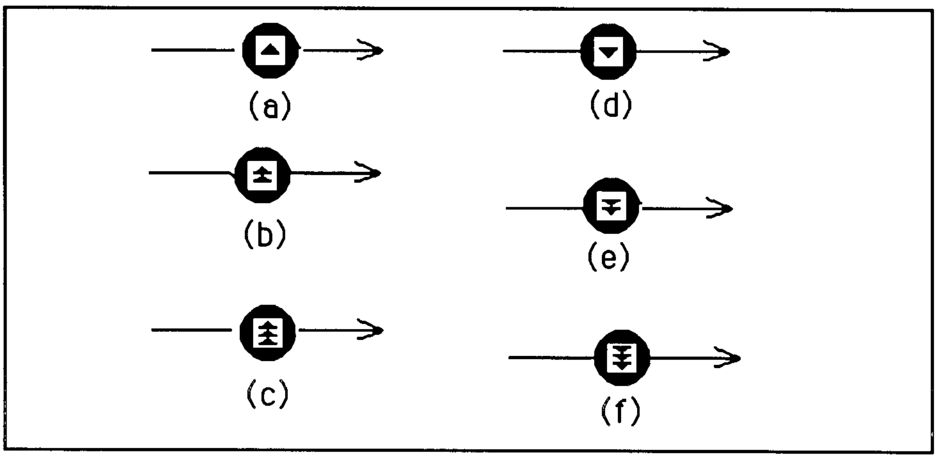


Figure 6.5: Different strengths and directions of cause of 'cumulative' Links: (a) same direction/ normal strength; (b) same direction/strong strength; (c) same direction/weak strength ; (d) opposite direction/ normal strength; (e)opposite direction/strong strength; (f)opposite direction/weak strength

- *Input Combination* - The input combination determines how one link will combine with the others in order to calculate the new value of the incoming variable. These inputs can be combined in three different ways (see Figures 6.6, 6.7):

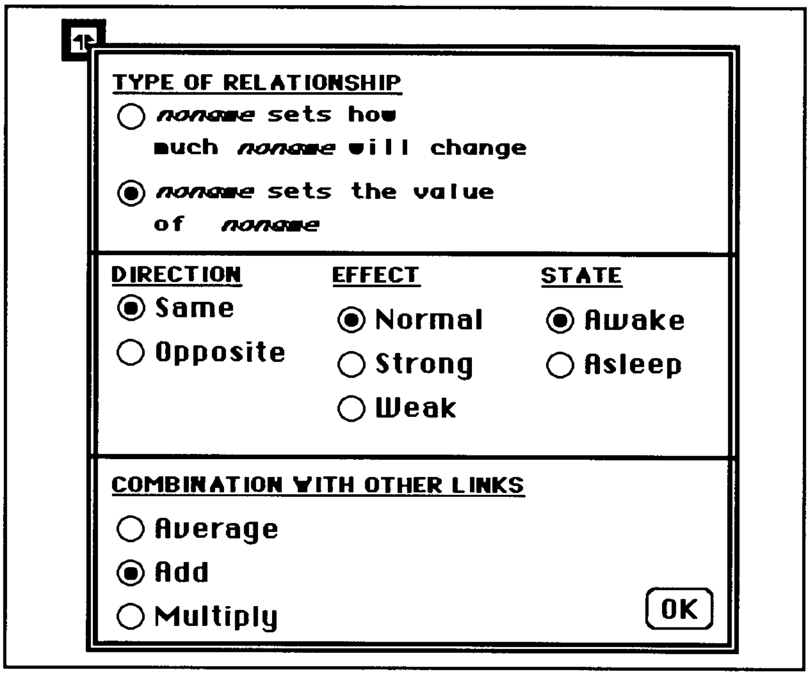


Figure 6.6: Information Box of a 'go together' Link

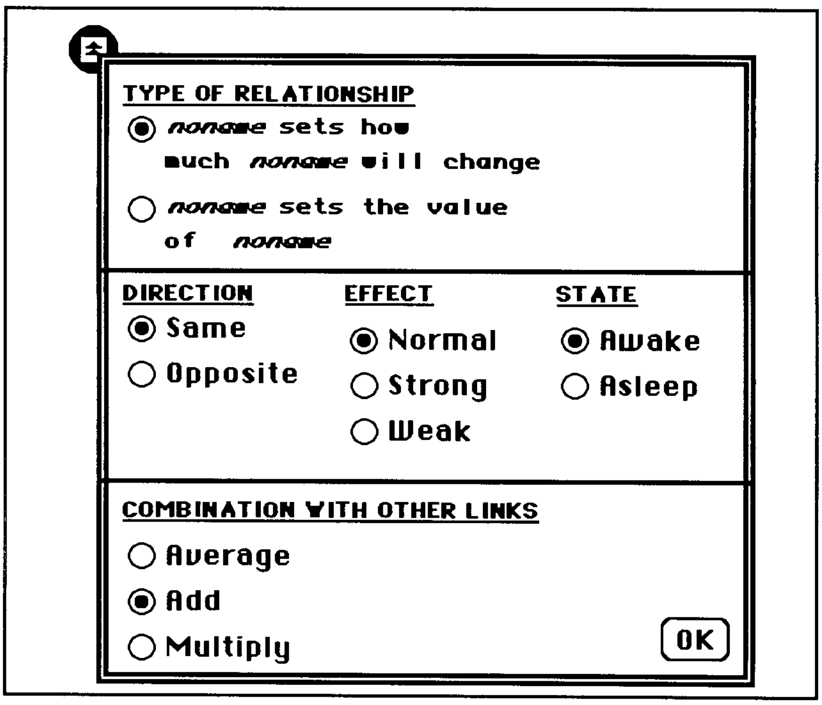


Figure 6.7: Information Box of a 'cumulative' Link

Average - Performs a weighted average of all links arriving in the same variable; *Additively* - Performs a weighted sum of all links arriving in the same variable; *Multiply* - Performs a weighted multiplication of all links arriving in the same variable. Other mathematical operations such as subtraction and division can be achieved by combining the input combination with opposite direction links.

To visually differentiate the different possible combinations of links the system uses two different patterns for the end of an arrow. A filled arrow (black triangle) indicates that the combination is multiplication. An empty arrow indicates that the combination is average or sum.

6.2.1.3 Modelling Possibilities and Examples

The introduction of new ways of combining inputs and the different possibilitites for the strength of links, permits LinkIt II to model algebraic relations and dynamic processes more accurately than LinkIt I.

Algebraic Models

'Go together' links with the new types of input combinations 'add' and 'multiplication', permits LinkIt II to represent algebraic relations in a more realistic way. Also the combination of these types of inputs with 'opposite direction' links, permits the system to perform 'subtraction' and 'division' respectively. Figure 6.8.a shows an example of calculating the total income of a waiter based on his/her *salary*, *tips* and *deductions* (subtracted from the other two inputs). Figure 6.8.b shows a special case of combining 'multiplication' with 'opposite' link to achieve the 'inverse' mathematical operation.

Boolean Models

Although the kinds of boolean models that can be constructed with LinkIt II are quite the same as LinkIt I (see Section 4.5.3.3), with this new version the user can choose one between three different possibilities for the output of an 'On/Off' variable (see also Figure 6.3): equal to 1('high'); equal to 0.25 ('low'); or follow the value of the variable. Figure 6.9 shows an example of the use of these three possibilities.

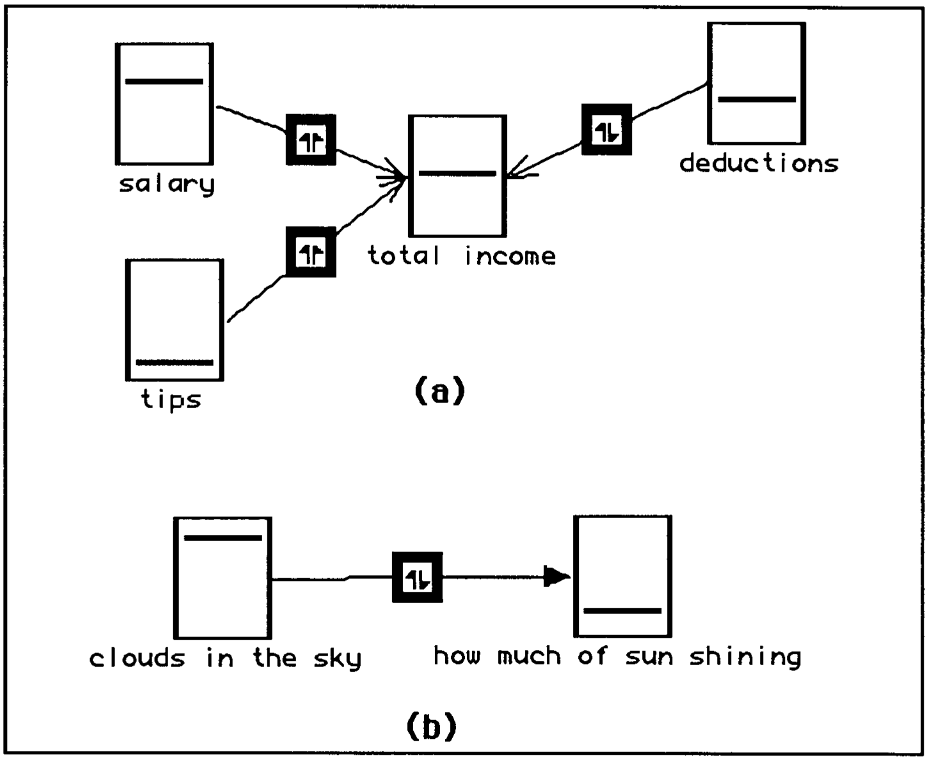


Figure 6.8: Two examples of algebraic models: (a) A model about the total income of a waiter where *deductions* is being subtracted from *tips* and *salary*; (b) A model about clouds and sun shining. *How much of sun shining* is the inverse of *clouds in the sky* (combining 'multiplication' and 'opposite' link).

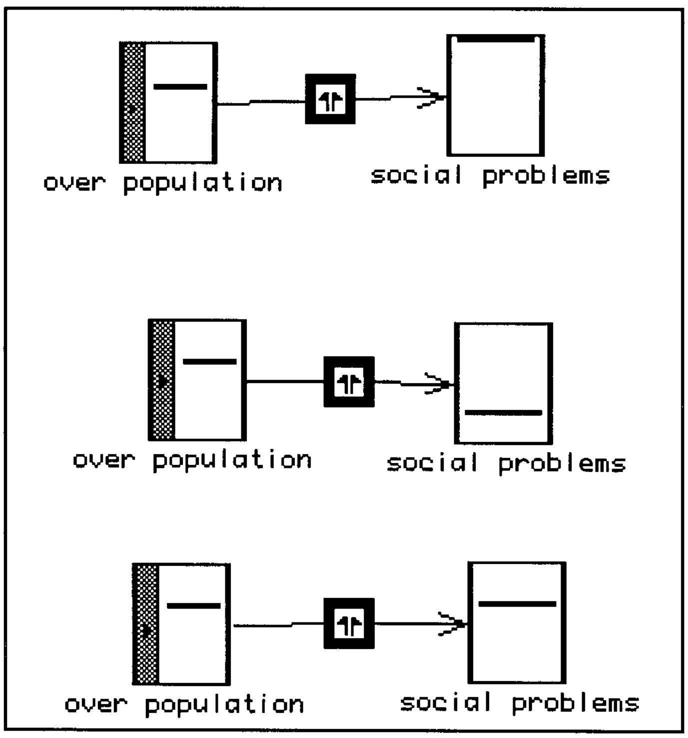


Figure 6.9: Three different values (and interpretations) for the output of an 'On/Off' variable. In the first case the output is set to 'high', in the second case it is set to 'low' and in the third case it is set to 'equal to'.

Differential Equations

The use of 'cumulative' and 'go together' links combined with the two types of existing variables provides an approximation to differential equations.Table 6.1 summarizes the different possibilitites of differential equations that can be modelled with LinkIt II followed by some examples on each of the categories mentioned in the Table.

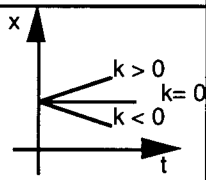
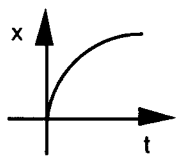
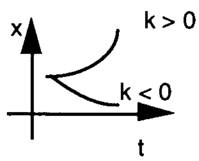
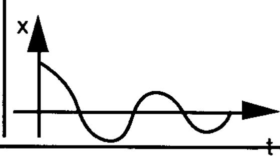
	Linear processes	Build up exponential	Exponential growth & decay	Oscillations
Ex. of problems	Pollution Population Fluids Eletromagnetic induction Etc.	Leaky tank Pollution LR circuit Etc.	Population Food absortion Nuclear decay RC circuits Etc.	Spring-mass system Pendulum Predator-prey Etc.
Dif. Equations	$dx/dt = +k$	$dx/dt = K*(M - x)$	$dx/dt = +Kx$	$m*\frac{d^2x}{dt^2} + b*\frac{dx}{dt} + kx = 0$
Order	1	1	1	2
Possible graphical solutions				

Table 6.1: A summary of dynamic modelling possibilities of LinkIt II

• Linear processes

Example (a) of Figure 6.10 shows a model about pollution of the air and its consequences on the health of the population and pollution control. In this model the rate of change of *pollution of the air* is the weighted sum of the values of *industries* and *cars*. So, mathematically speaking the relation between them can be expressed as:

$$\frac{d(\text{Pollution of the air})}{dt} = +K$$

where K = $\underline{W_1 * \text{Industries} + W_2 * \text{Cars}}$

Example (b) is another case of linear process. However, at this time the rate of change of *car-rust* is determined by the multiplication of the values of *oxygen* and *water*. The use of 'multiplication combination' here fits well to represent the chemical rule: "rust can only occur if water and oxygen are present". Also *market price* was defined as being inversely proportional to the amount of *car-rust* . So when *car-rust* is very high the *market price* of the car is very low.

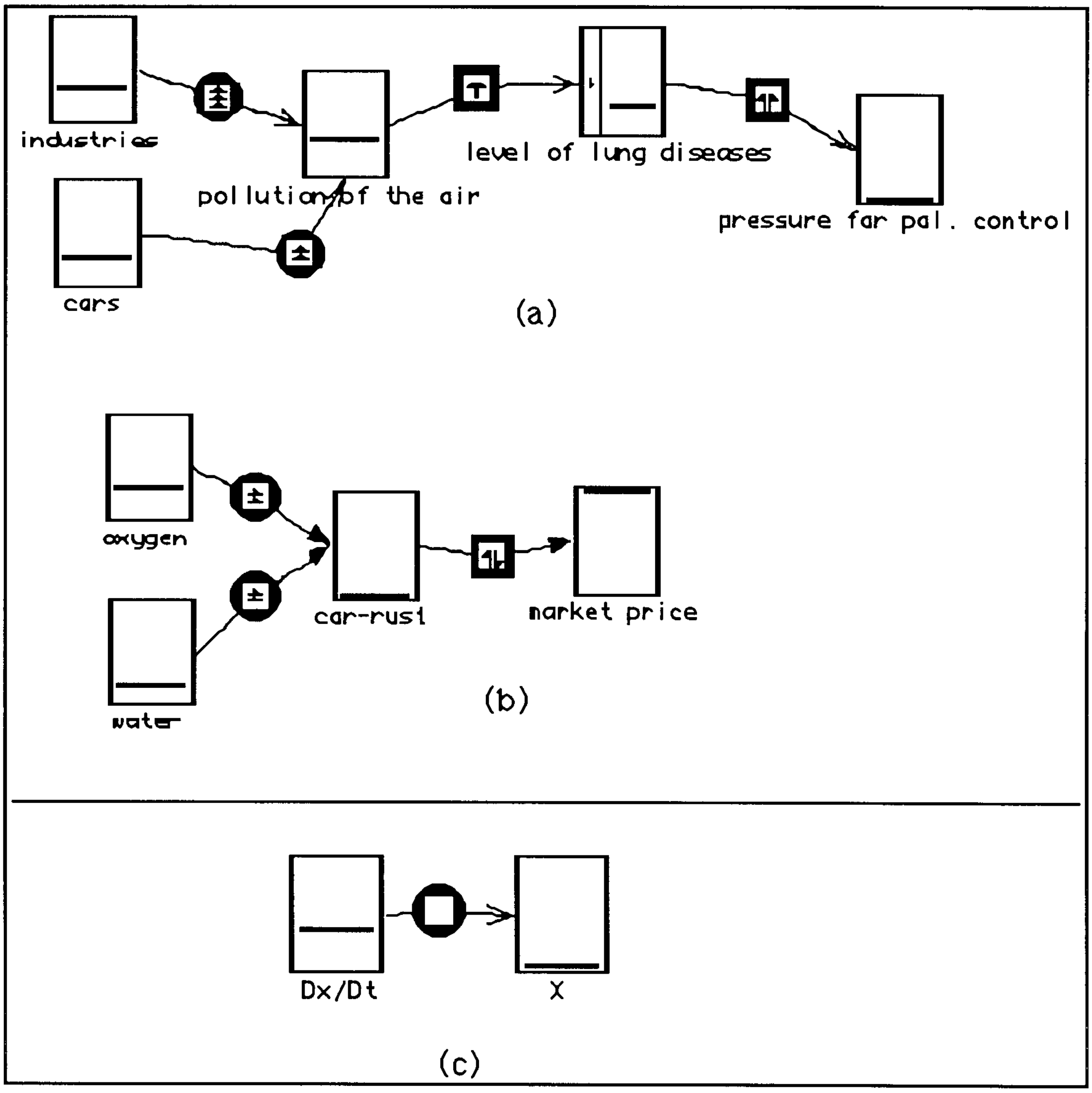


Figure 6.10: Two examples of linear process ((a) and (b)) and its general representation with LinkIt II ((c)). The different signs of the links (in (c)) determine the value of the constatnt K in the equations.

• <u>Build-up exponential process</u>

The general behaviour of this kind of process is its tendency to adjust (or to approximate) to a certain desired value represented by the constant **M**. Figures 6.11 and 6.12 show a very simple example about the behaviour of the eye-pupil. In both examples the variable box *normal light level* represents the desired (or comfortable) light level for a certain species, which is exactly the constant **M** mentioned above. Figure 6.11 shows the adjustment of the *eye-pupil* from below and Figure 6.12 shows its adjustment from above.

Although this example could also be modelled by LinkIt I, here it is more accurate because this new version of the system permits variables to be added.

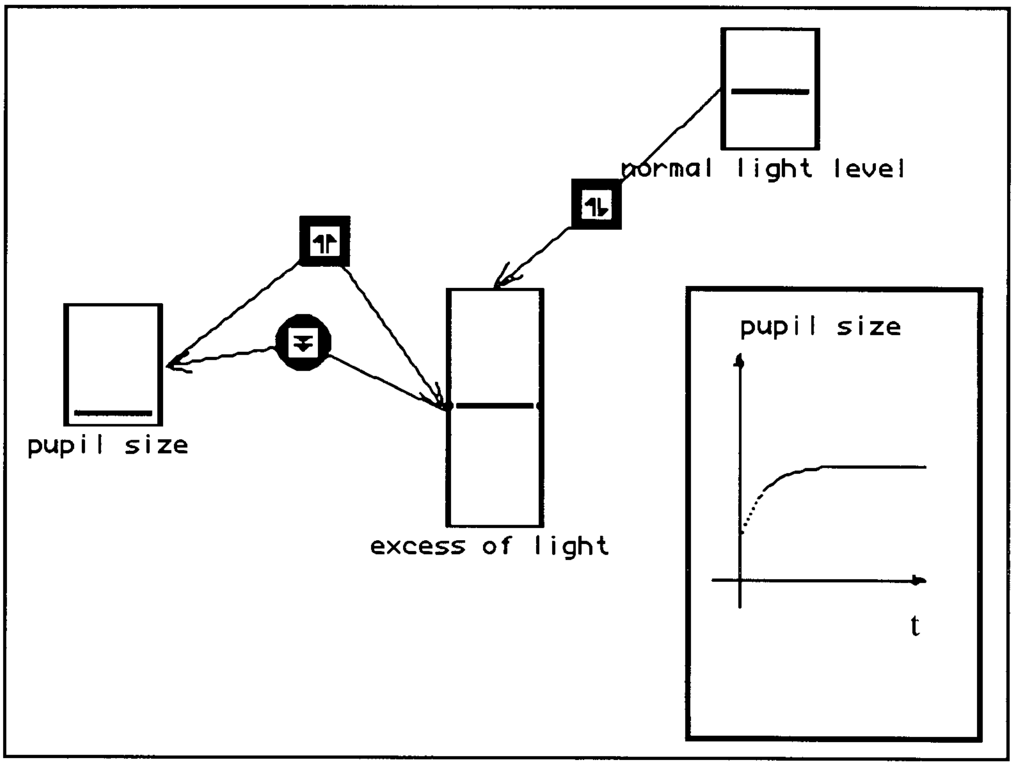


Figure 6.11: Build-up exponential process: An example of a simple model about eye-pupil. The values of the variables are the initial conditions before the simulation. The variable *pupil size* started with a value smaller than *normal light level*.

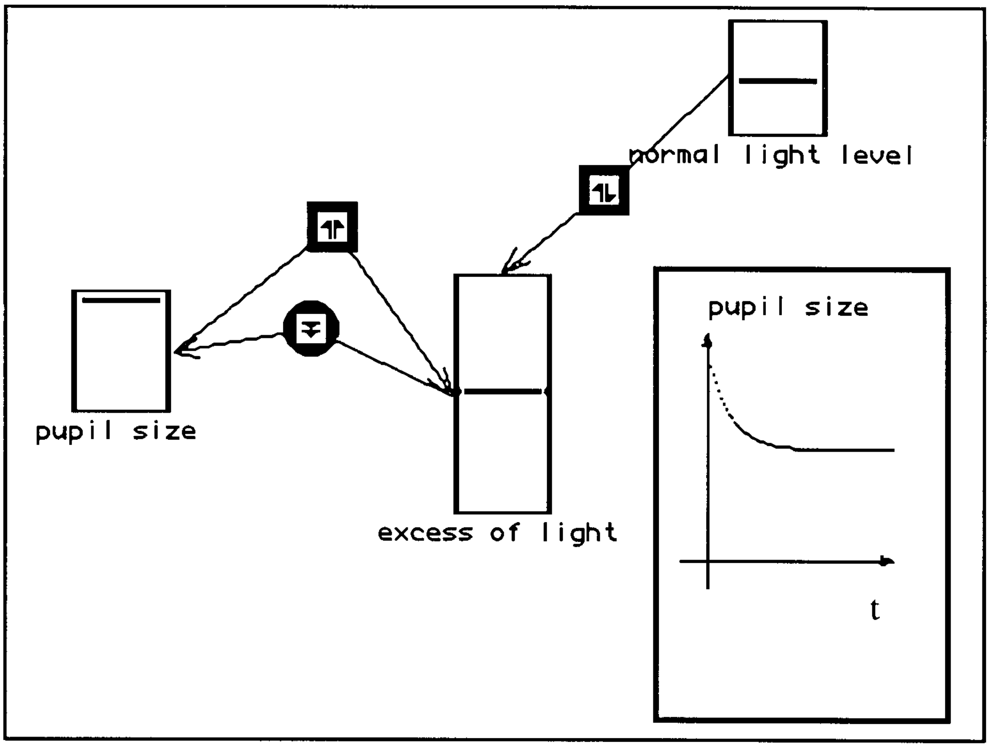


Figure 6.12: Build-up exponential process: An example of a simple model about eye-pupil. The values of the variables are the initial conditions before the simulation. The variable *pupil size* started with a value bigger than *normal light level*.

• Exponential growth/decay

Exponential processes can be seen as a particular case of build-up exponential where the constant **M** (see respective equations in Table 6.1) does not exist.

Exponential growth/decay can be achieved with LinkIt II by simply combining 'cumulative' and 'go together' links in a feedback loop (see Figure 6.13).

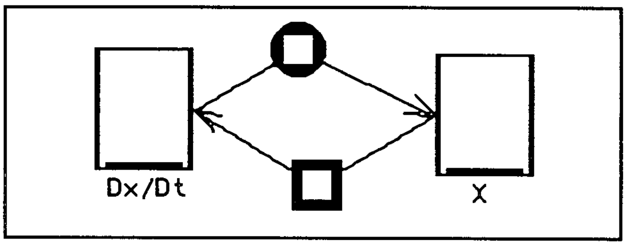


Figure 6.13: General structure of an exponential process with LinkIt II. The different signs of the links determine the value of the constant K in the equations

The example shown in Figure 6.14 uses the combination 'multiplication' to calculate the value of *food supply* and suggests that the *money needed* to keep someone alive grows exponentially. This is not what happens in the "real world" ![2].

The example shown in Figure 6.15 is a case of exponential decay which is achieved by setting the variable *births* and its links to 'sleep'.

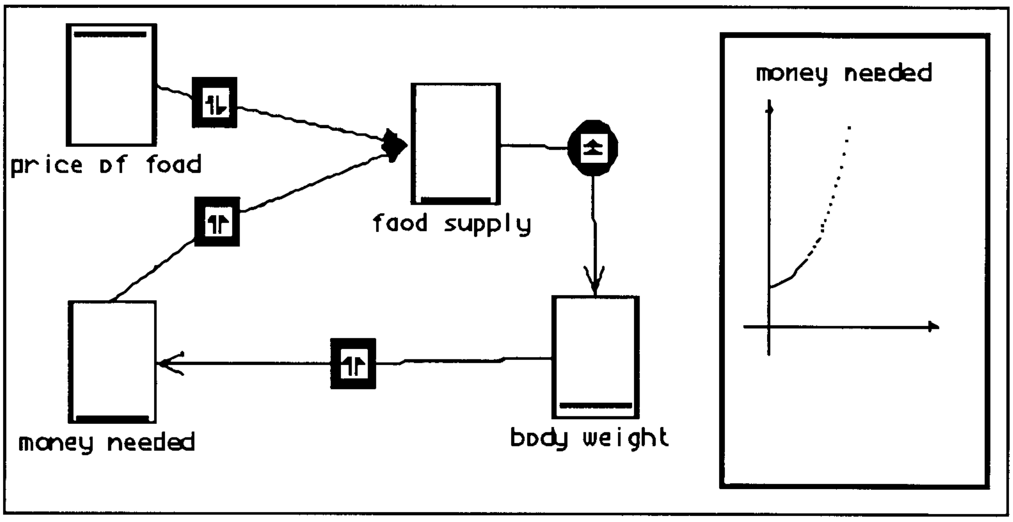


Figure 6.14: An incomplete model about body growth producing an exponential growth

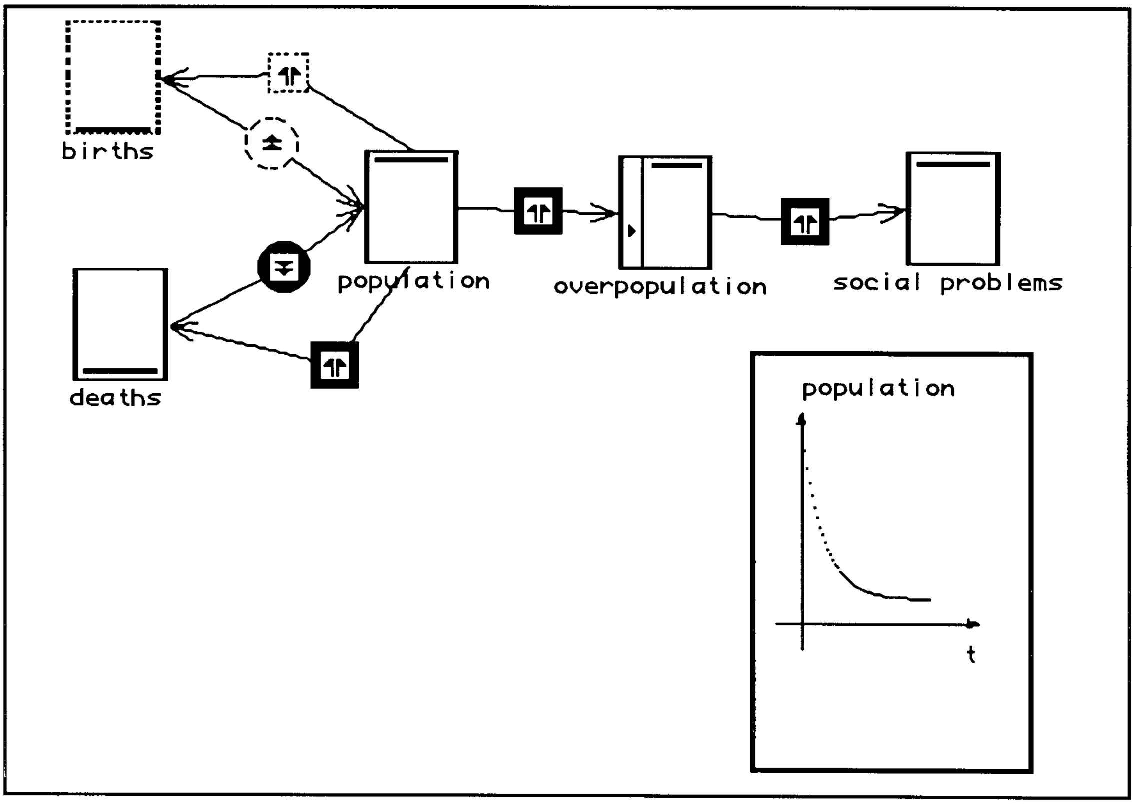


Figure 6.15: Example of a population model. In this case exponential decay can be achieved by setting the variable births 'asleep'.

[2] The problem here is that a fundamental part of the problem about consumption of energy is missing. If it was there neither *body weight* nor *money needed* would increase exponentially.

Another way of achieving exponential growth/decay is by setting the parameter 'change by itself'. Figure 6.16 gives an example of this case comparing a leaky tank with a normal one.

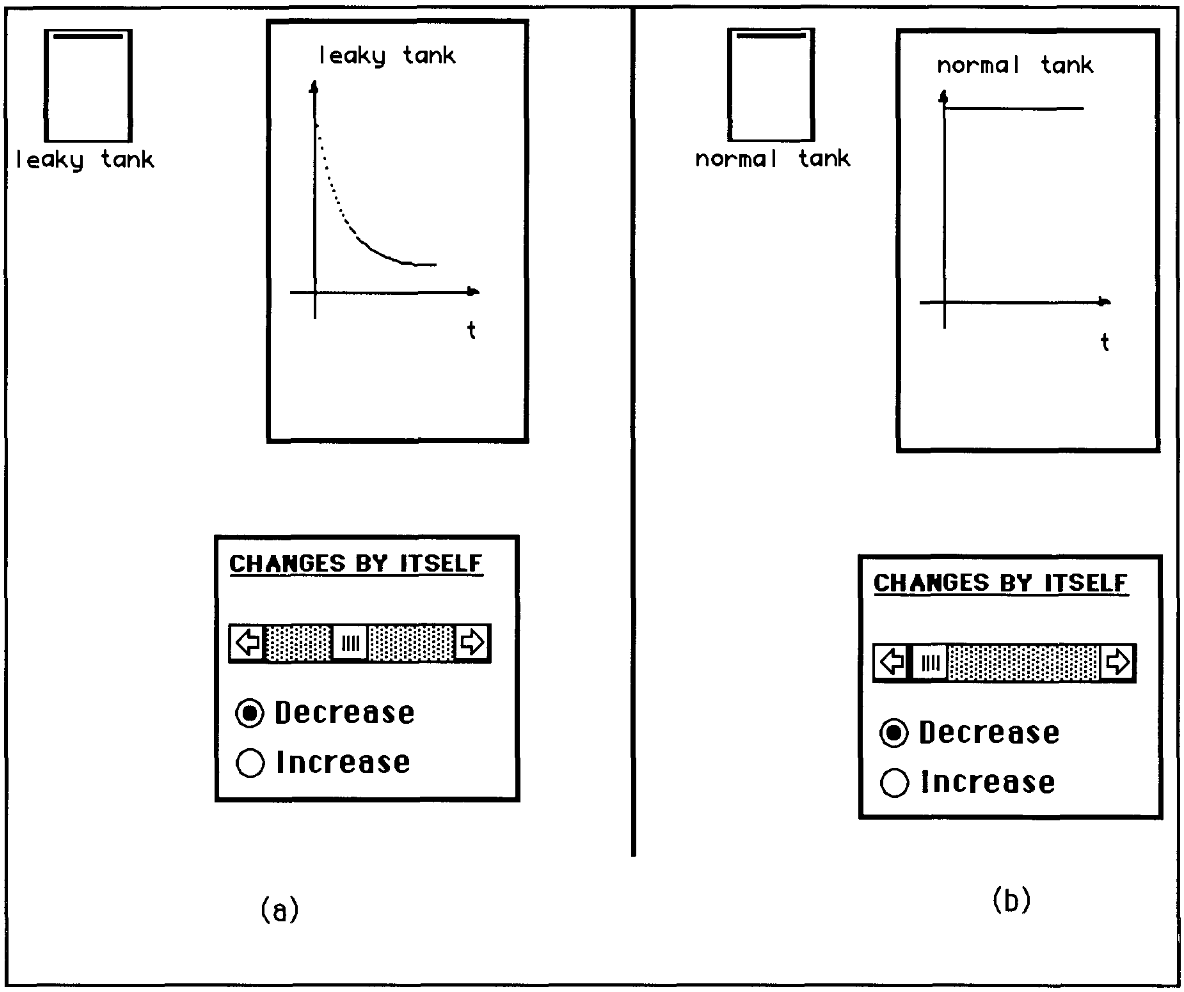


Figure 6.16: Exponential decay/growth can also be achieved using the 'changes by itself' parameter. Case (a) has this parameter set to decay. Case (b) is reset (no changes)

• <u>Oscillation</u>

Many natural and social systems exhibit a certain kind of oscillatory behaviour. Most of these phenomena can be represented with LinkIt II. The example shown in Figure 6.17 is about an harmonic oscillator.

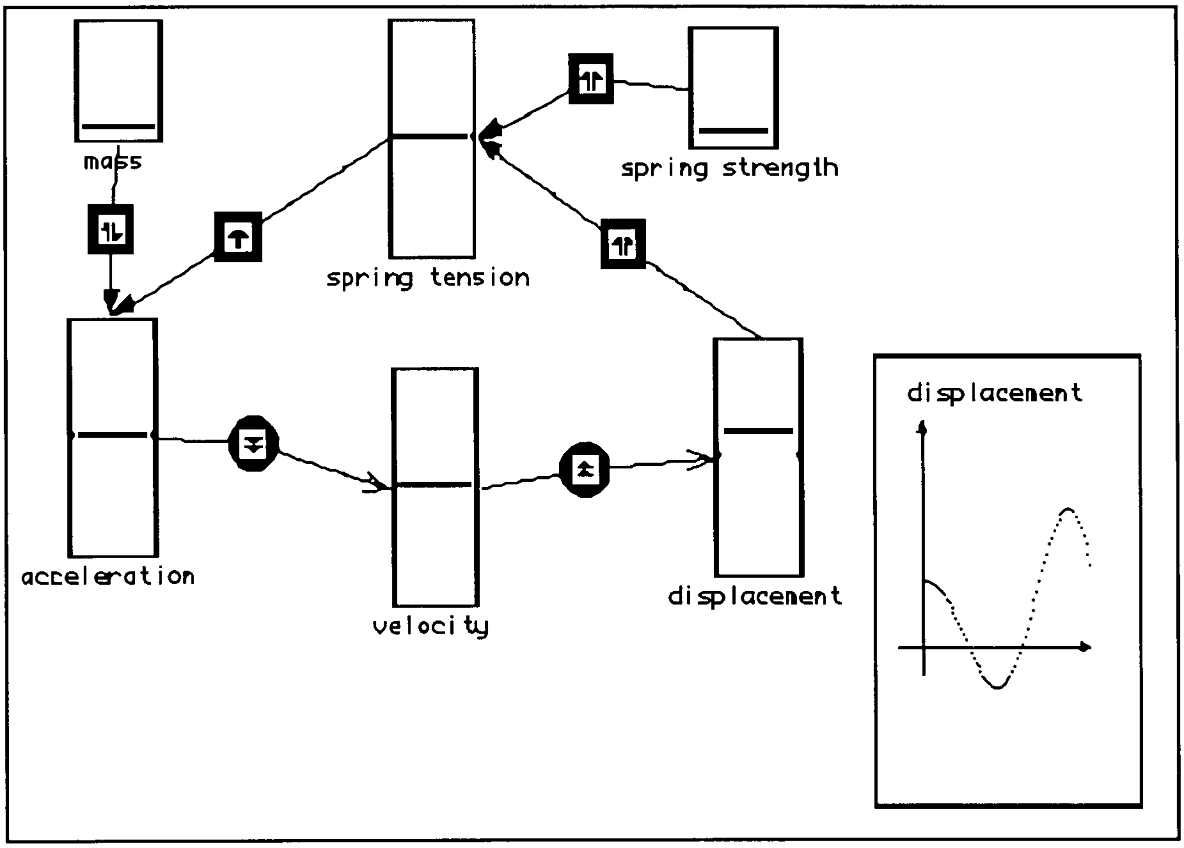


Figure 6.17: Example of an Harmonic oscillator[3]

It is important to notice here that second order derivatives can be achieved by combining 'cumulative' links. In Figure 6.17 the relation between *acceleration* and *displacement* is representing exactly this case.

6.2.2 The Desktop

The desktop of LinkIt II (see Figure 6.18) is very similar to but simpler than LinkIt I. It also contains two areas (system area and working area) and the control panel is located in the same place. Some graphical buttons that were presented in the previous version are absent, some new ones were created and others were redesigned. The menu bar, located above the top of the window, now has one menu only (called Model)[4].

These changes were made to incorporate new functionality to the system and to permit a less modal style of interaction[5] between the user and the system when he/she is creating a model or playing with it.

---

[3] In this model, the amplitude increases with time. This is due to the Euler approximation in effect used in the calculation.

[4] In fact two menus are presented: Model and Options. However the Options menu is automatically inserted by cT environment when the system is performing certain operations and it is not part of LinkIt environment.

[5] A modal interface can be defined as a style of interface in which certain operations are only available when the system is in a certain mode (or state). So interfaces can be classified as less or more modal according to the number of operations related to a certain mode. In principle the less modal an interface is, the easier the software is manipulated by the user (it gives more freedom to the user).

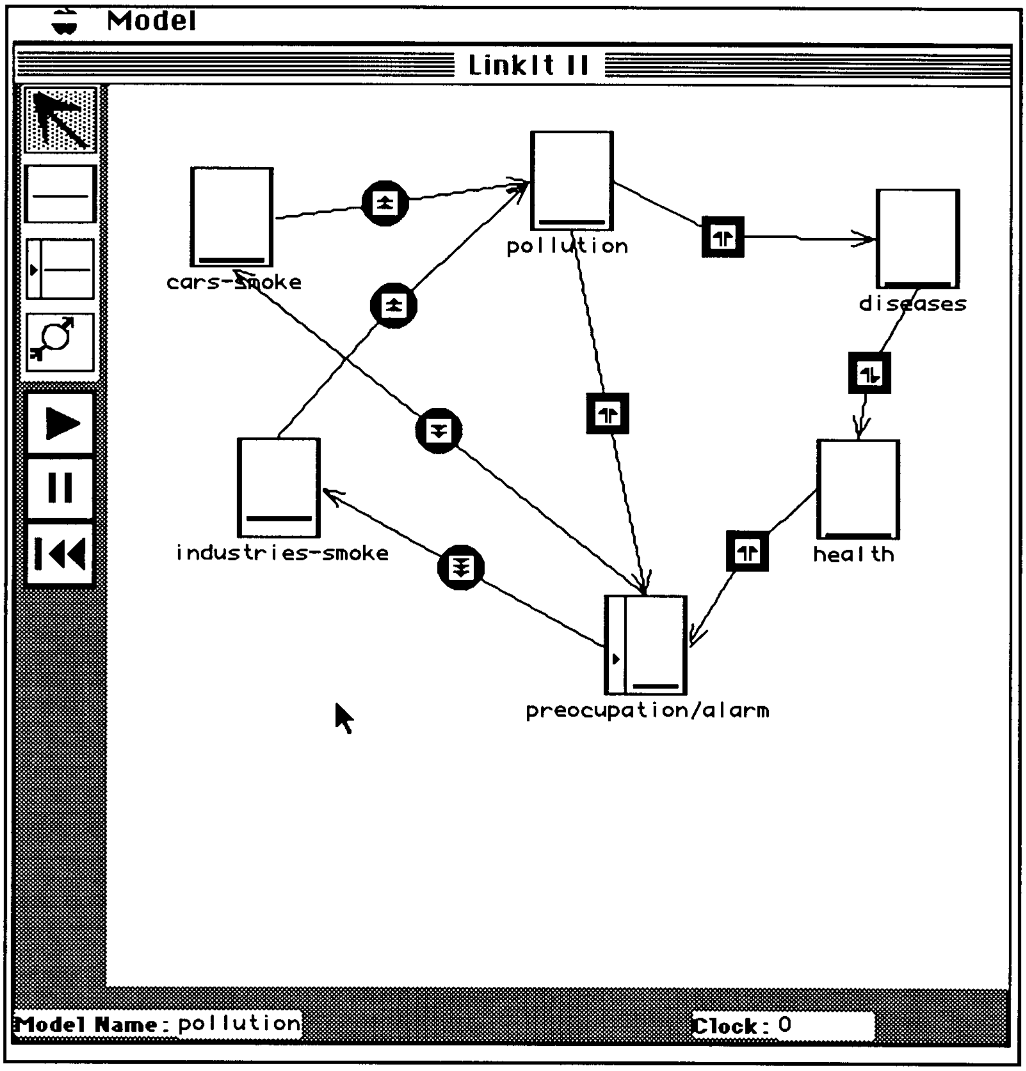


Figure 6.18: LinkIt's window with a model being constructed

6.2.3 <u>The Control Panel</u>

The control panel (see Figure 6.19) has 7 buttons divided into two functionally different groups. The first group has four buttons which are related to the creation of a model (editing operations). They represent the basic manipulations provided by the system and the three different building blocks one can use to create a model. The second group has three buttons and is related to the execution of a model .

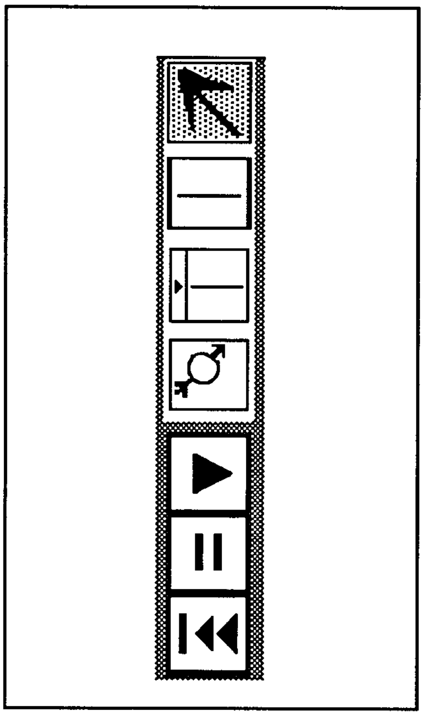


Figure 6.19: Control Panel

6.2.4 Operating the System

Although the interface operations (see Chart 6.2) presented in this version are the same as those in Prototype I, the way the user operates the system is different. Now none of the editing operations have precedence over the other classes of operations. The system always returns to a state of “waiting for a user’s action”, characterised by the shape of the cursor (an arrow pointing up and left) and the first icon (an arrow) in the Control Panel selected.

According to the action taken by the user the system executes a certain operation and comes back to the state of “waiting for a user’s action”. This change was made to make the user-system interaction less modal and therefore easier to manipulate (Shneiderman, 1992) and to avoid certain problems pointed out by the Preliminary study (see Section 5.4).

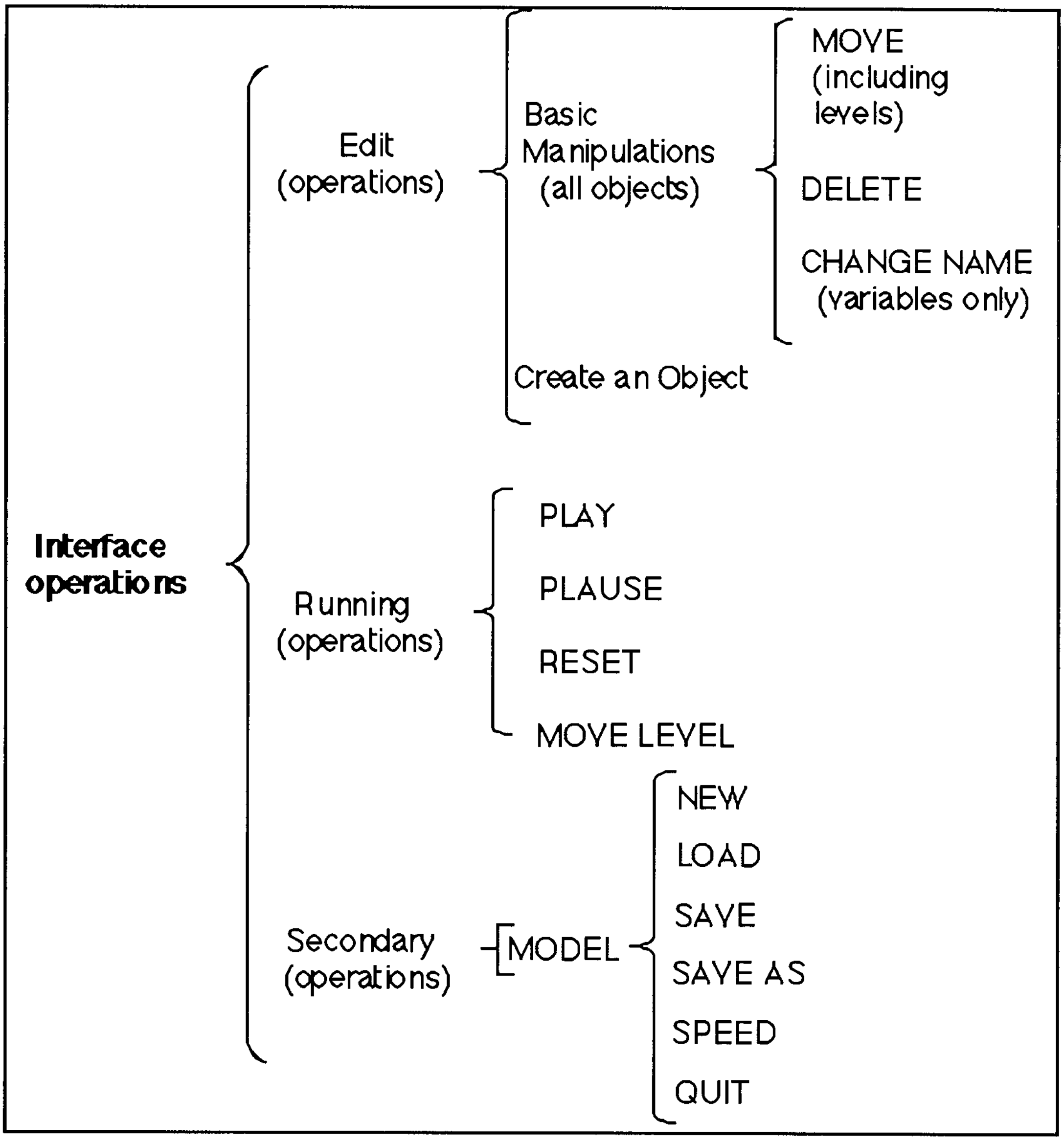


Chart 6.2: Interface operations

6.2.4.1 Description of the Editing Operations

Create an Object

To create a new object the user has to select the corresponding button from the control panel (single click on it) and another single click on the place where he/she wants to create the new object. If the object is a link, after selecting the appropriate button in the control panel, the user has to single click on each of the pairs of variables he/she wants to connect. After creating the object, the system comes back to the state of "waiting for a user's action".

When a button to create an object is selected, the cursor shape changes its appearance, resembling the object selected.

Basic Manipulation - Move and Delete

The Basic manipulations MOVE (variables, links and levels) and DELETE are performed in the same way as Prototype I (see Section 4.5.4.1).

Basic Manipulation - Change the name

To change the name of a variable, the user has to single click anywhere on the name. An editing box is automatically opened on the screen with the name of the variable on it, close to that variable. The user can change the name in a way very similar to the editing mode of a word processor. In the same way as the Information box, the system will accept the new name after pressing the OK button.

Change the properties (object-variable)

This manipulation is performed in the same way as in Prototype I (see Section 4.5.4.1)

Change the properties (object-link)

In order to change the properties of a link, the user has to double click on the hot part of the link (the small circle/square) which evokes an Information Box with all properties of the link. After changing them, the user has to click on the OK button.

6.2.4.2 Description of the Running Operations (Related to the Execution of the Model)

Apart from the graph operation (not used in this version), the running operations are the same as in Prototype I. The difference is basically the visual presentation of the buttons in the Control Panel. They were modified to have a closer resemblance of the control buttons of HI-FI and VCR systems (see Figure 6.19 and compare with Figure 4.1).

An important modification introduced here is the possibility given to the user to change the value of a variable while the system is running. When a simulation is in progress (run button is pressed) and the system detects that the user is attempting to move a level of a certain variable-box, it puts itself in a pause state and waits for the user to finish moving the level. Afterwards it resumes the running process.

6.2.4.3 Description of the Secondary Operations

This version has just one group of secondary operations (*Menu* operations) associated with a pop down menu - called *ModeL'* - in the menu bar.

All operations in this group are performed in the same way as in Prototype I (see Section 4.5.4.3). However, a new operation was introduced : *Speed* . It permits the user to see the model running in slow motion or at normal speed (default option). When it is selected, a Dialogue box is presented with two toggle buttons. After selecting the appropriate one, the user has to press the OK button (see Figure 6.20).

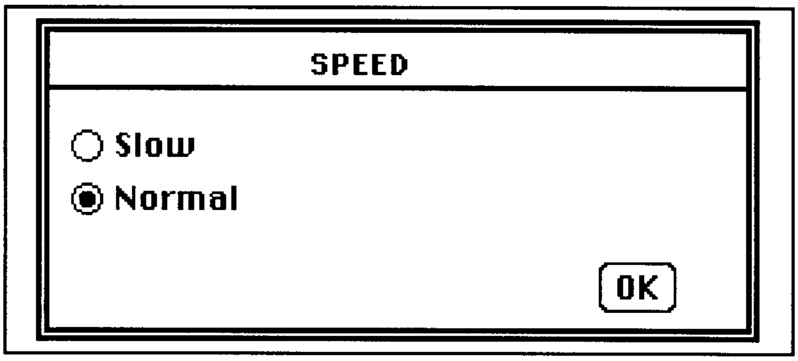


Figure 6.20: Dialogue box to set the speed of simulations

6.3 THE INFERENTIAL ENGINE OF THE SOFTWARE

The inferential engine of LinkIt is always hidden from the user and is responsible for (1) mathematically interpreting the model created by the user; (2) making it evolve over time; and (3) giving, on the screen, a visual feedback of the model evolving over time. These 3 components are presented and discussed in the two subsections that follow and examples are given in 6.3.3.

6.3.1 Mathematics of the Software

The mathematics that control the behaviour of the model when it is running is, like Prototype I, adapted from the computations that govern some neural networks. However in this case it is more self consistent permitting the values of the variables to vary in a linear regime.

The possible values of all variables lie in the interval [-1,1]. However because in this version new ways of combining links and new behaviours of 'On/Off' variables were implemented it was decided to use new squashing and activation functions (see chapter 9 and 10 of (Rumelhart & McLelland, 1987)) to govern their behaviours, making computations in a linear regime (range $-\infty$, $+\infty$) squashing them for display, and expanding them again for calculation.

Squashing Function

The squashing function used in this Prototype has the following general form: $(e^x - 1) / (e^x + 1)$. One of its advantage is that it has a quasi-linear output for small input values which makes it possible to present the 'addition' and 'multiplication' combinations in a quite reasonable way (see Figure 6.21). Another important factor is that most computer languages provide algorithms to compute the exponential function and its inverse (ln(x)).

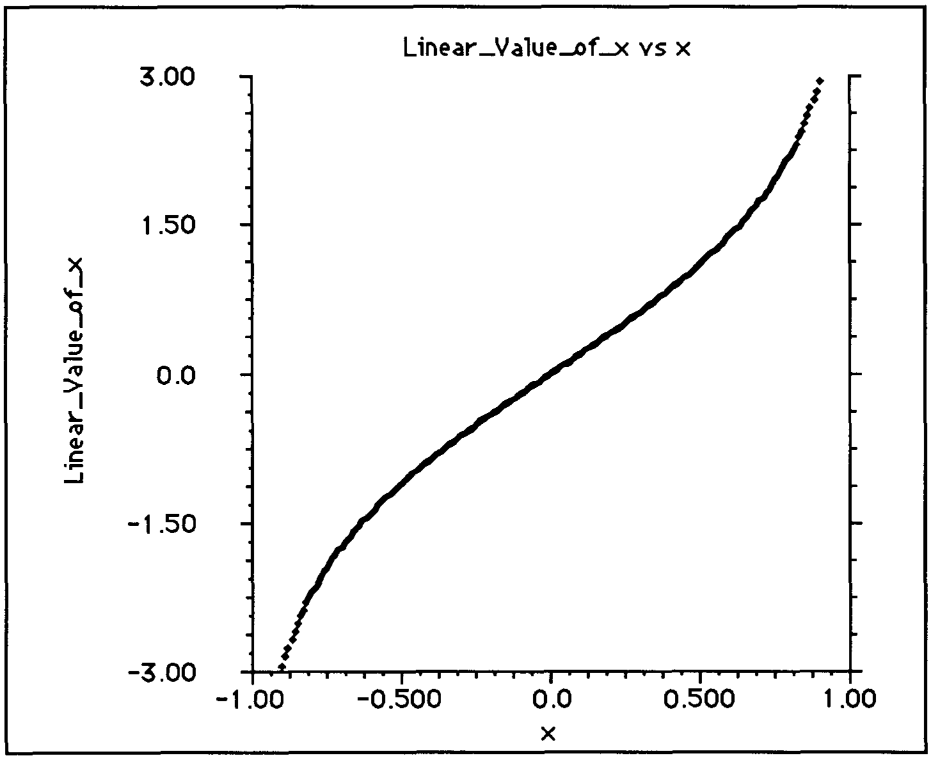


Figure 6.21: Graph of Linear_Value (x) = $Log_n((1+x)/(1-x))$

Activation Function

Activation functions are used to control the behaviour of On/Off variables. Basically the system provides three different outputs for this type of variable. The first one (see Figure 6.22) is when the user sets the output combination of a variable to be 'equal to' its amount level. Here after triggering, the system just passes the current value of the variable to the outgoing link(s)[6].

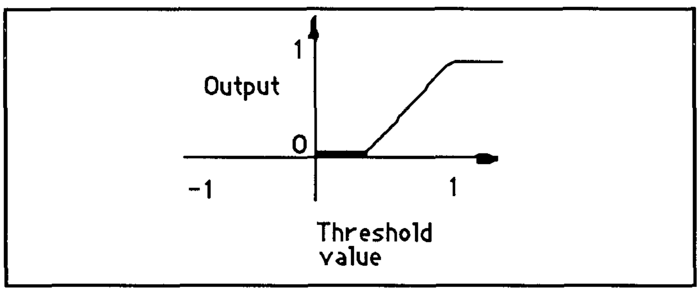


Figure 6.22: Activation function with output equal to the amount level

[6] If the On/Off variable is operating with values below zero the activation function gives an opposite value in relation to the one shown in Figure 6.22.

The second possibility is to have a 'high' output when the variable is triggered (see Figure 6.23). This is done by setting the output combination of a variable to maximum (option high). In this case the value passed to the outgoing link(s) is equal to 1.

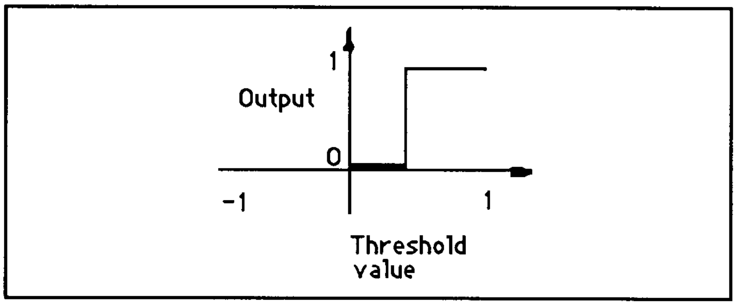


Figure 6.23: Activation function with output maximum

The third possibility is to have a low output (see Figure 6.24) when the variable is triggered (option 'low' in the Information box of the variable). The value passed through the outgoing link(s) is equal to 0.25.

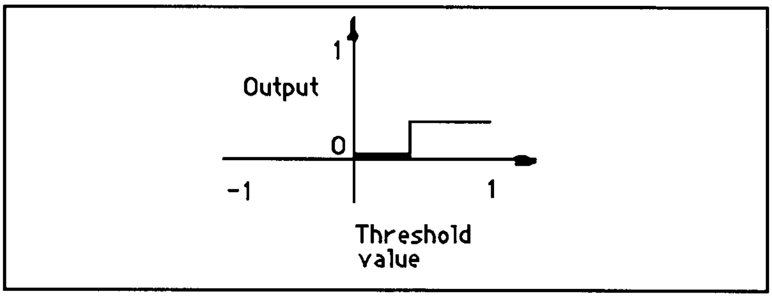


Figure 6.24: Activation function with output minimum

The computational mathematics of LinkIt was developed in such a way that the possible values for its variables are within the interval [-1, 1]. So the minimum possible value for a variable is -1, 0 is the normal value and +1 the maximum possible value.

The computational mathematics for each type of variable of the system is presented below.

<u>Notation used</u>

If we consider that a certain model has n variables $a_j (1 <= j <= n)$ and L links (L >= 0) with weights w, then a link between a_i and a_j can be represented by w_{ij}. Let m be the number of links arriving at a_j, $a_j(t)$ the internal (squashed) value of a variable j at time t, $Damping_j$ a 'changes by itself' value associated to it, VL_i the value of a variable a_i after being expanded to the linear regime, $Input_j$ the calculated value of the input combination of a certain variable a_j, and $Squash_j$ the squashing function which restricts the value of the variable (after recalculating its new value) to the interval [-1, 1].

The calculation of a value of a dependent variable in the range [-1, 1] is performed in three steps:

(I) Expand input values to a linear regime (calculate VL_i) using

$$VL_i = \ln((1 + a_i)/(1 - a_i))$$

(II) Form appropriate weighted value of inputs, combining them as specified (calculate $Input_j$).

(III) Squash computed values to the non-linear domain [-1, 1], using

$$Squash_j = \exp(a_j-1)/\exp(a_j+1)$$

The squashing and expansion functions are, of course, each the inverse of the other. A fourth step is to convert the values calculated in (III) to screen displays.

Steps (I) and (III) are the same for all dependent variables. Step (II) varies according to the type of input combination of a variable, and in detail, will be:

• In the case the input combination is *average*, the **Input** will be (1):

$$\mathbf{Input_j} := \left(\sum_{i=1}^{m} (VL_i * w_{ij}) / \sum_{i=1}^{m} w_{ij}\right) \qquad \text{with } i <> j$$

• In the case the input combination is *adding*, the Input will be (2):

$$\mathbf{Input_j} := \left(\sum_{i=1}^{m} (VL_i * w_{ij}\right) \qquad \text{with } i <> j$$

• In the case the input combination is *multiplication*, the Input will be (3):

$$\mathbf{Input_j} := \left(\prod_{i=1}^{m} (VL_i * w_{ij})\right) \qquad \text{with } i <> j$$

(where w_{ij} is negative, the weight is interpreted as division).

After calculating the **Input** for a certain variable a_j the system calculates its new value according to the type of its incoming links:

• 'Cumulative' links (4):

$$DT = Constant := 0.1$$
$$VL_j(t-1) = \ln((1 + a_j(t-1))/(1 - a_j(t-1)))$$
$$VL_j(t) := VL_j(t-1) + DT * Input_j - Damping_j * VL_j(t-1)$$ [7]
$$Squash_j := (\exp(VL_j(t)) - 1) / (\exp(VL_j(t)) + 1)$$
$$a_j(t) := Squash_j$$

• 'Go together' links (5):

$$VL_j(t-1) = \ln((1 + a_j(t-1))/(1 - a_j(t-1)))$$
$$VL_j(t) := Input_j - Damping_j * VL_j(t-1)$$
$$Squash_j := (\exp(VL_j(t)) - 1) / (\exp(VL_j(t)) + 1)$$
$$a_j(t) := Squash_j$$

6.3.2 <u>The Running Process</u>

At the beginning of a simulation, the system creates a matrix that contains information about the dependent variables that are involved in at least one ('awake') relationship. The independent variables that have an associated 'changes by itself' value different from zero are stored in another vector and are treated separately.

At each iteration the system chooses, for each dependent variable, the appropriate groups of equations (described above) and recalculates them. At this time the system pays special attention to the 'On/Off' variables that have not been triggered yet. In these cases, these variables will put zero in their output.

After recalculating the value of a variable, the system checks whether it is an appropriate value for that variable or not. For instance, in the case of a bigger than zero variable, if its new calculated value is negative then its new value will be approximated to zero.

The values used in each iteration are the values of the object-variables in the previous iteration. During the first iteration (t=0) the initial value for each object-variable is that set by the user.

---

[7] The first part of this equation [$VL_j(t) := VL_j(t-1) + DT * Input_j$] is in reality Euler's numerical method used to solve ordinary differential equations given a set of initial conditions.

At the end of an iteration the system checks which variables have to be updated on the screen. For each of these variables it applies an offset function that converts its internal value to a screen value. After that it updates the variable-boxes and the clock on the screen and goes to check whether any interruption occurred. In cases where the user tried to change the value of a variable while the model is iterating, the new value set to that variable box will become its new "initial" value and it will be used by the simulation module when it resumes.

6.3.3 Some Examples

In order to make clear how the system works I shall present some of the different possibilitites of LinkIt II in two different examples. The first example is about pollution of the air and the second example presents a simplified version of the well known problem about predator and prey.

6.3.3.1 A Model About Pollution of the Air

Suppose someone created the model shown in Figure 6.25.

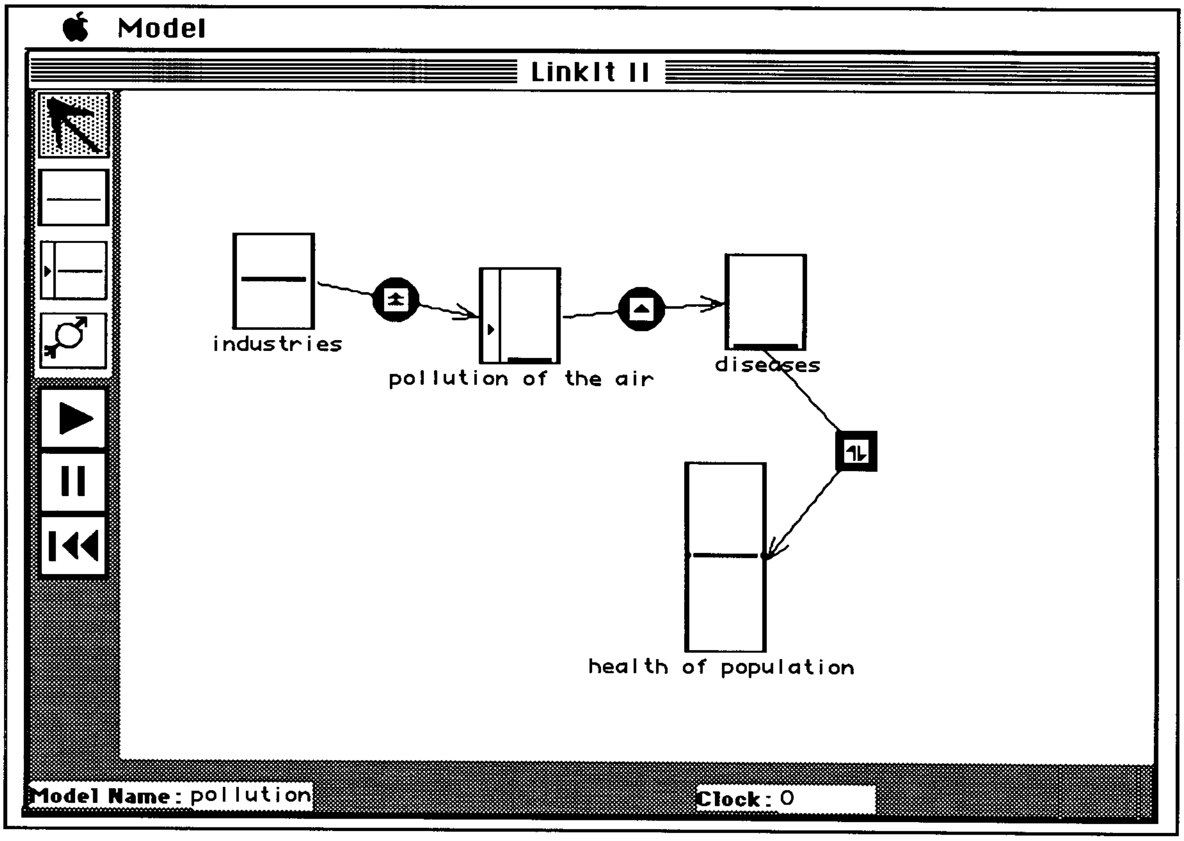


Figure 6.25: Model about pollution. The variables are set by the user before a simulation.

At the beginning of the simulation the initial (internal) values of the variables corresponding to what was set by the user are:

Industries: Smooth variable; 'bigger than zero'; no 'changes by itself' associated; independent variable with internal value = 0.52 .

Pollution of the air: On/Off variable; 'bigger than zero'; set to 'trigger when (the amount level becomes) above' the threshold level; output set to 'equal to' the amount level; no 'changes by itself' associated; dependent variable with 1 incoming 'normal-same direction-cumulative' link; amount level (AL) = 0.00; trigger level (TL) = 0.33 .

Diseases: Smooth variable; 'bigger than zero'; no 'changes by itself' value associated, dependent variable with 1 incoming 'weak-same direction-cumulative' link; initial internal value = 0.0 .

Health of the population: Smooth variable; 'any value'; no 'changes by itself' value associated, dependent variable with 1 incoming 'normal-opposite direction- go together' link; internal initial value = 0.0 .

As can be seen, the model shown above has no feedback loop. So after some iterations (in this case 52) the model will reach a stable configuration where the levels of the dependent variables will reach one of the extremes of their boxes.

<u>Calculating the next value of *industries*:</u>

Because *industries* is an independent variable with no associated 'changes by itself' value, it is not going to take part in the simulation process. In other words, no information about *industries* will be created in the structures employed by the simulation module.

<u>Calculating the next value of *pollution of the air*:</u>

Because *pollution* has one ingoing 'cumulative' link with input combination set to 'adding', the equations presented in Section 6.3.1 employed to calculate its next value are (2) and (4).

The only input of *pollution* is *industries* which is a 'smooth' variable. So it is necessary first to calculate its corresponding linear value in order to calculate the value of the Input of *pollution of the air* (called here $\mathbf{Input_{pollution}}$). In order to do so the set of equations (2) has to be used.

$VL_{industries} = \ln((1+industries)/(1-industries)) = \ln((1+0.52)/(1-0.52)) = 1.15$

$$\textbf{Input}_{pollution} = \left(\sum_{i=1}^{1} (VL_{industries} * w_{ij}) = (1.15 * 1) = 1.15\right.$$

Now it is possible to calculate the next internal value of *pollution of the air* ($Pollution_{(t=1)}$):

$DT = 0.1$

$VL_{pollution(t=0)} = \ln(1/1) = 0.$

$VL_{pollution(t=1)} = VL_{pollution(t=0)} + DT * Input_{pollution} - Damping_{pollution} * VL_{pollution(t=0)} =$

$0.0 + 0.1 * 1.15 - 0.0 * 0.0 = 0.115$

$Squash_{pollution} = (\exp(VL_{pollution(t=1)}) - 1)/(\exp(VL_{pollution(t=1)}) + 1) =$

$(\exp(0.115) - 1)/(\exp(0.115)+1) = (1.121 - 1)/(1.121 + 1) = 0.05$

$Pollution_{(t=1)} = Squash_{pollution} = 0.05$

According to table 6.2 only after the sixth iteration will the calculated value of the amount level of *pollution of the air* become bigger than the value of its threshold (TL = 0.33). So until the end of that iteration its output will be set to zero. In this way, the variable *diseases* - which is dependent of *pollution of the air* - will have a value equal to zero coming through its incoming link until the end of that iteration.

Clock	***Industries***	***Pollution of the air***		***Diseases***	***Health of pop.***
1	0.52	0.05	0.33	0.0	0.0
2	0.52	0.11	0.33	0.0	0.0
6	0.52	0.33	0.33	0.0	0.0
7	0.52	0.38	0.33	0.01	0.0
8	0.52	0.43	0.33	0.03	-0.01
51	0.52	0.99	0.33	0.99	-0.99
52	0.52	0.99	0.33	0.99	-0.99

Table 6.2: Values of the variables involved in a simulation of the model about pollution shown in Figure 6.25. The first column of *pollution of the air* corresponds to the value of its amount level. The second column corresponds to its trigger level.

From the end of the sixth iteration on *pollution of the air* will become ON. To indicate that, the left part of its variable box will be highlighted (see Figure 6.26).

<u>Calculating the next value of *diseases*:</u>

Like *pollution of the air, diseases* will employ the same set of equations (2) and (4). The only difference is the value of the 'weight' of its incoming link which is, in this case, equal to 0.5 (it is an 'weak-same direction' link). However at this first iteration the output of *pollution* is zero which makes the value of $\mathbf{Input_{diseases}}$ = 0.0 and its next value (when t=1) equal to zero.

At the end of the sixth iteration the amount level of *pollution* equalizes its trigger level. So during the next iteration (t=7)*diseases* will have an Input <> 0.

$VL_{pollution}$ = ln((1+pollution)/(1-pollution)) = ln ((1+0.33)/(1-0.33)) = 0.69

$$\mathbf{Input_{diseases}} = \quad (\sum_{i=1}^{1} (VL_{diseases} * w_{ij}) = (0.69 * 0.5) = 0.34$$

Delta = 0.1

$VL_{diseases(t=6)}$ = ln(1/1) = 0.

$VL_{diseases(t=7)}$= $VL_{diseases(t=6)}$ + DT * $Input_{diseases}$ - $Damping_{diseases}$ * $VL_{diseases(t=6)}$ =

0.0 + 0.1 * (0.34) - 0.0 * 0.0 = 0.034

$Squash_{diseases}$ = (exp($VL_{diseases(t=7)}$) - 1)/(exp($VL_{diseases(t=7)}$) + 1) =

(exp(0.034) - 1)/(exp(0.034)+1) = (1.034 - 1)/(1.034 + 1) = 0.01

$Diseases_{(t=7)}$ = $Squash_{diseases}$ = 0.01

<u>Calculating the next value of *health of the population*:</u>

Health of the population has just one incoming 'go together-opposite direction' link. So the sets of equations used here are (2) and (5).

The type of the incoming link implies that *health of the population* has to follow its input - *diseases* - although in an opposite way.

Also because *diseases* and *health of the population* were both set to zero in the beginning of the simulation and the value of *diseases* did not change until the seventh iteration, then the value of *health of the population* will remain the same until the end of that iteration. During the eighth iteration *diseases* will have an output different from zero (see Table 6.2) which will cause a change in the internal value of *health of the population.* The following equations refer to the calculation of the eighth iteration of this variable.

$VL_{diseases}$ = ln((1+diseases)/(1-diseases)) = ln ((1+0.01)/(1-0.01)) = 0.02

$$\textbf{Input}_{\textbf{health}} = \left(\sum_{i=1}^{1} (VL_{diseases} * w_{ij}\right) = (0.02 * (-1)) = -0.02$$

$VL_{health(t=8)}$= $Input_{health}$ - $Damping_{health}$ * $VL_{health(t=7)}$ = (-1.02) - 0.0 * 0.0 = -1.02

$Squash_{health}$ = (exp($VL_{health(t=8)}$) - 1)/(exp($VL_{health(t=8)}$) + 1) =

(exp(-0.02) - 1)/(exp(-0.02)+1) = (0.98- 1)/(0.98 + 1) = - 0.01

Health of $population_{(t=8)}$ = $Squash_{health}$ = - 0.01

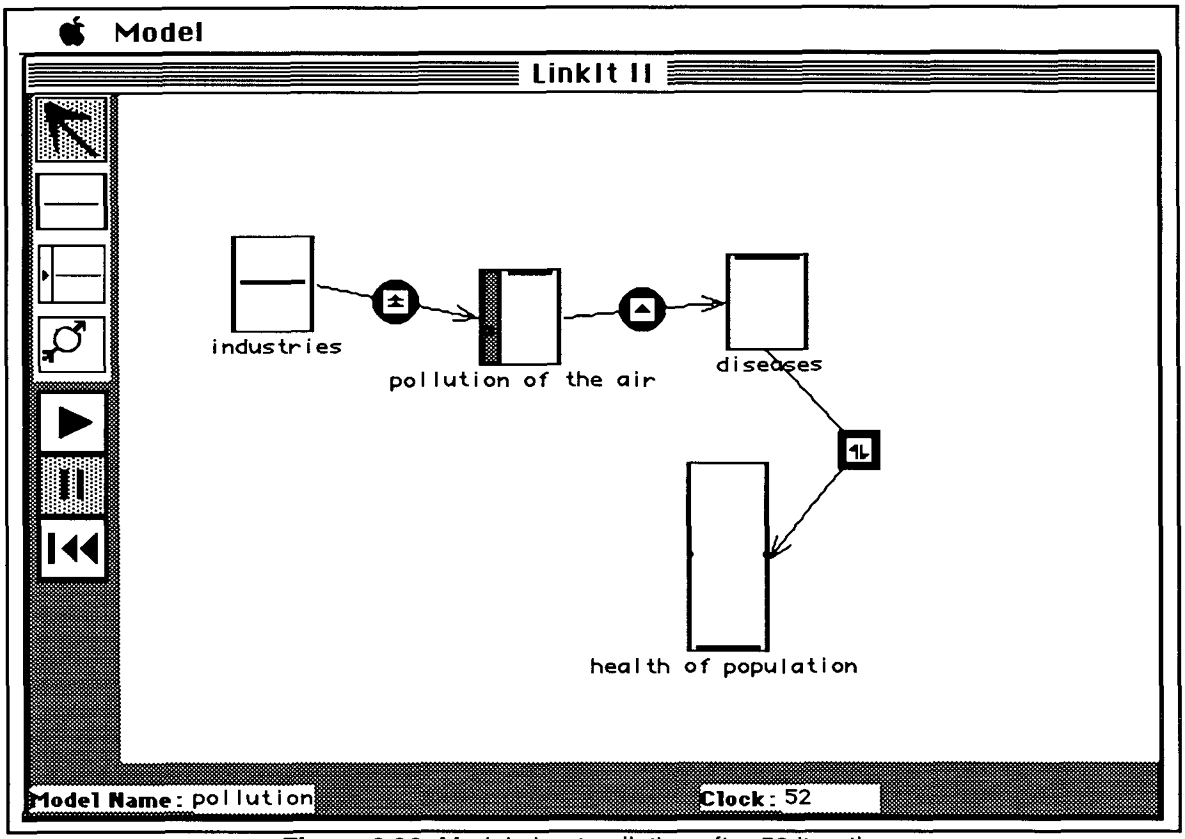


Figure 6.26: Model about pollution after 52 iterations

<u>Presenting the new values on the screen:</u>

After recalculating the internal value of a certain variable the system has to transform it to a "screen value". Before doing so, some approximations are needed to avoid round-off errors imposed by the finite precision arithmetic of the computer language and division by zero. The first of these approximations is to consider that if the module of a certain variable $|x| \geq |0.99|$ then $|x|$ will be approximated to $|0.99|$. The second is that in case $|x| \leq |0.001|$ then $|x|$ will be approximated to 0.0.

To transform the internal value of a variable to a screen value, the system employs the following offset function:

Screen_Value(x) = INT(1+ 42 * (1 - Value(x)))

After that it compares this new screen value with the old screen value of the variable to see whether any change has occurred. If the test is positive this new information is stored in a vector that will be used to update the all variable-boxes on the screen at the same time at the end of each iteration.

6.3.3.2 A Model About Predator-Prey

This second example is a simple implementation about the classical problem of predator-prey. Figure 6.27 shows the initial conditions of the problem. Compared to the previous example, there are two new features being used here: (1) the input combination of *rabbits killed* is 'multiplication' and (2) *rabbits* and *foxes* have 'change by itself' values associated to them to represent respectively the ideas of " new rabbits being born" and "foxes dying" (see Figure 6.28). Also in this case there exist 2 feedback loops.

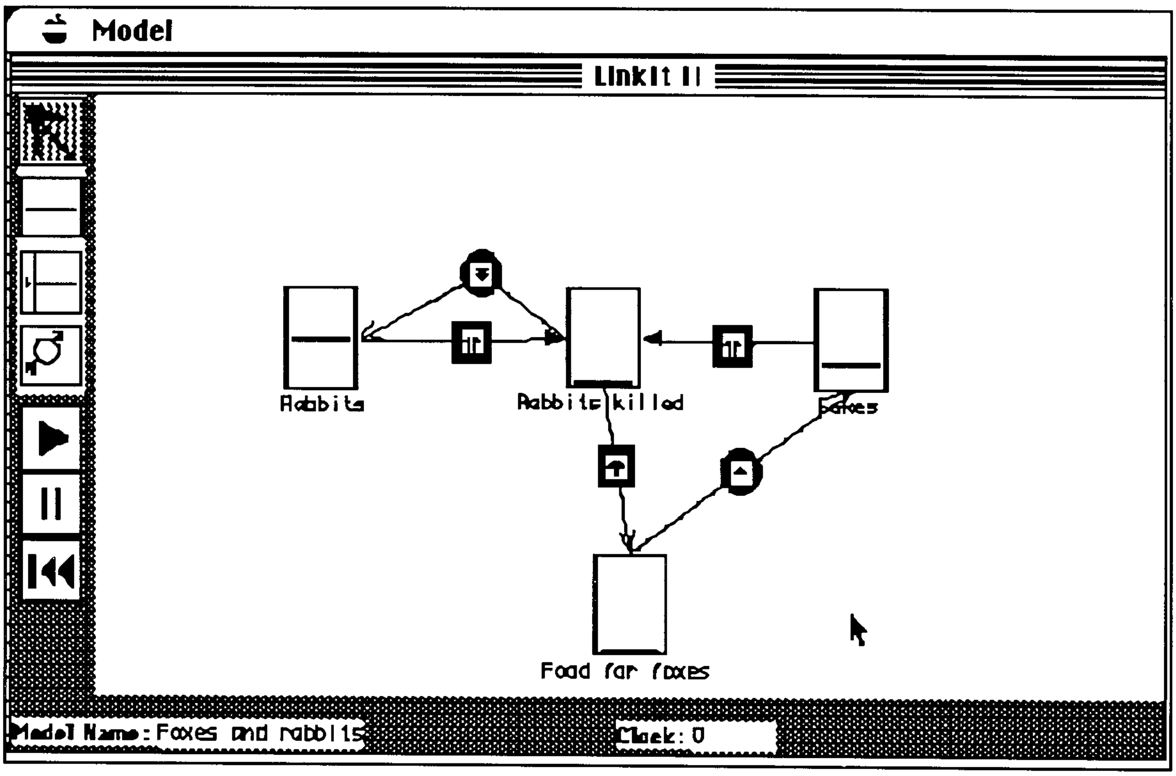


Figure 6.27: Model about predator-prey before simulation

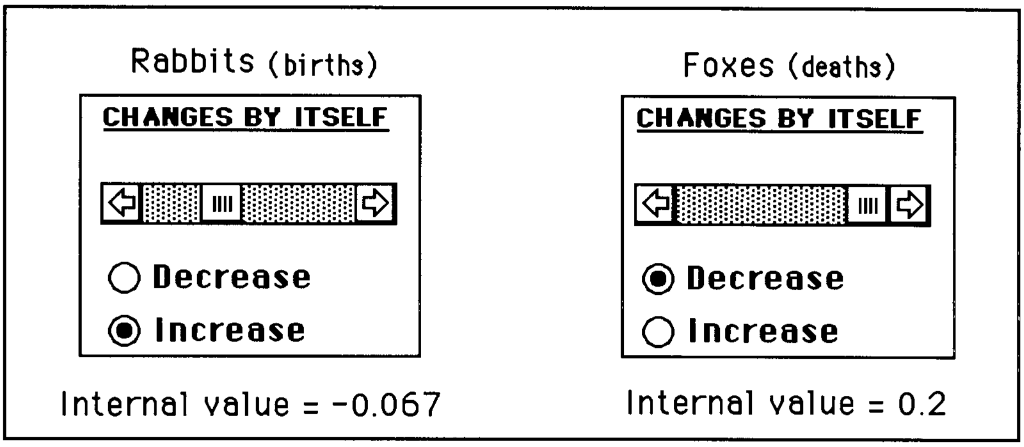


Figure 6.28: 'Changes by itself' values set to *rabbits* and *foxes*

At the beginning of the simulation the initial (internal) values of the variables corresponding to what was set by the user are:

Rabbits: Smooth variable; 'bigger than zero'; 'changes by itself' value associated; dependent variable with 1 incoming 'cumulative opposite direction normal' link; internal value = 0.50;

Rabbits killed: Smooth variable; 'bigger than zero'; no 'changes by itself' value associated; dependent variable with 2 incoming 'go together same direction normal' link; internal value = 0.0;

Foxes: Smooth variable; 'bigger than zero'; 'changes by itself' value associated; dependent variable with 1 incoming 'cumulative same direction weak ' link and 1 outgoing 'go together same direction normal' link ; internal value = 0.23;

Food for foxes: Smooth variable; 'bigger than zero'; no 'changes by itself' value associated; dependent variable with 1 incoming 'go together same direction normal' link and 1 outgoing 'cumulative same direction weak' link ; internal value = 0.0.

As can be seen, the model shown above has 2 feedback loops. So, it is very difficult to predict whether it will reach a stable configuration before running it. Below,only the calculation of the variables for the first iteration is shown. Table 6.3 presents a summary of the variables until 132 iterations and Figure 6.29 presents the model in the 132th iteration.

Calculating the next value of *rabbits* :

Rabbits is a dependent variable with an associated 'changes by itself' value and 1 ingoing 'cumulative' link (from *rabbits killed*). So the set of equations to be used are (2) and (4):

$VL_{rabbits\ killed} = \ln((1+0)/(1-0)) = 0$

$$\mathbf{Input}_{\mathbf{rabbits}} = \left(\sum_{i=1}^{1} (VL_{rabbits\ killed} * w_{ij}) = (0.0 * (-1)) = 0.0\right.$$

$VL_{rabbits(t=0)} = \ln((1+ rabbits_{(t=0)})/(1- rabbits_{(t=0)})) = \ln((1 + 0.5)/(1-0.5)) = 1.09$
$VL_{rabbits(t=1)} = VL_{rabbits(t=0)} + DT * \mathbf{Input}_{\mathbf{rabbits}} - Damp_{rabbits} * VL_{rabbits(t=0)} \Rightarrow$
$VL_{rabbits(t=1)} = 1.09 + 0.1*0.0 - (-0.067)*1.09 = 1.163$
$Squash_{rabbits} = (\exp(VL_{rabbits(t=1)}) - 1)/ (\exp(VL_{rabbits(t=1)}) + 1) = 0.52$
$Rabbits_{(t=1)} = Squash_{rabbits} = 0.52$

Calculating the next value of *rabbits killed* :

Rabbits killed is a dependent 'smooth' variable with 2 ingoing 'go together' links being multiplied. So the set of equations to be used are (3) and (5):

$VL_{rabbits} = \ln((1+0.5)/(1-0.5)) = 1.09$
$VL_{foxes} = n((1 + 0.23)/(1-0.23)) = 0.468$

$$\mathbf{Input}_{\mathbf{rabbits\ killed}} := \left(\prod_{i=1}^{2} (VL_i * w_{ij})\right) = (1.09*1) * (0.468*1) = 0.51$$

$VL_{rabbits\ killed(t=0)} = \ln((1+ rabbits\ killed_{(t=0)})/(1- rabbits\ killed_{(t=0)})) \Rightarrow$
$VL_{rabbits\ killed(t=0)} = \ln((1 + 0)/(1-0)) = 0.0$
$VL_{rabbits\ killed(t=1)} = \mathbf{Input}_{\mathbf{rabbits\ killed}} - Damp_{rabbits\ killed} * VL_{rabbits\ killed(t=0)} \Rightarrow$
$VL_{rabbits\ killed(t=1)} = 0.51 - 0.0 * 0.0 = 0.51$
$Squash_{rabbits\ killed} = (\exp(VL_{rabbits\ killed(t=1)}) - 1)/ (\exp(VL_{rabbits\ killed(t=1)}) + 1) \Rightarrow$
$Squash_{rabbits\ killed} = 0.259$
$Rabbits\ killed_{(t=1)} = Squash_{rabbits\ killed} = 0.26$

Calculating the next value of *foxes* :

Foxes is a dependent variable with an associated 'changes by itself' value and 1 ingoing 'cumulative' link (from *food for foxes*) So the set of equations to be used is also (2) and (4):

$VL_{food\ for\ foxes} = \ln((1+0)/(1-0)) = 0$

$$\textbf{Input}_{\textbf{foxes}} = \quad (\sum_{i=1}^{1} (VL_{food\ for\ foxes} * w_{ij}) = (0.0 * (0.5)) = 0.0$$

$VL_{foxes(t=0)} = \ln((1+ foxes_{(t=0)})/(1- foxes_{(t=0)})) = \ln((1 + 0.23)/(1-0.23)) = 0.468$
$VL_{foxes(t=1)} = VL_{foxes(t=0)} + DT * \textbf{Input}_{\textbf{foxes}} - Damp_{foxes} * VL_{foxes(t=0)} =>$
$VL_{foxes(t=1)} = 0.468 + 0.1*0.0 - (0.2)*0.468 = 0.374$
$Squash_{foxes} = (\exp(VL_{foxes(t=1)}) - 1)/ (\exp(VL_{foxes(t=1)}) + 1) = 0.185 = 0.19$
$Foxes_{(t=1)} = Squash_{foxes} = 0.19$

Calculating the next value of *food for foxes* :

Food for foxes is a dependent 'smooth' variable with just 1 ingoing 'go together' link. So the set of equations to be used here are (2) and (5):

$VL_{food\ for\ foxes} = \ln((1+0.0)/(1-0.0)) = 0.0$

$$\textbf{Input}_{\textbf{food for foxes}} := (\sum_{i=1}^{2} (VL_i * w_{ij})) = 0.0 * 1 = 0.0$$

$VL_{food\ for\ foxes(t=0)} = \ln((1+ food\ for\ foxes_{(t=0)})/(1- food\ for\ foxes_{(t=0)})) =>$
$VL_{food\ for\ foxes(t=0)} = \ln((1 + 0)/(1-0)) = 0.0$
$VL_{food\ for\ foxes(t=1)} = \textbf{Input}_{\textbf{food for foxes}} - Damp_{food\ for\ foxes} * VL_{food\ for\ foxes(t=0)} =>$
$VL_{food\ for\ foxes(t=1)} = 0.0 - 0.0 * 0.0 = 0.0$
$Squash_{food\ for\ foxes} = (\exp(VL_{food\ for\ foxes(t=1)}) - 1)/ (\exp(VL_{food\ for\ foxes(t=1)}) + 1) =>$
$Squash_{food\ for\ foxes} = 0.0$
$Food\ for\ foxes_{(t=1)} = Squash_{food\ for\ foxes} = 0.0$

Presenting the new values on the screen:

The same set of rules used in the previous example has to be used here to present a certain variable on the screen: (1) Perform the approximations to avoid “rounding problems” and division by zero; (2) Apply an offset function to calculate the screen value of a certain level; (3) Update all variable-boxes at the same time on the screen; (4) Update the clock.

Clock	*Rabbits*	*Rabbits killed*	*Foxes*	*Food for foxes*
1	0.52	0.26	0.19	0.0
29	0.98	0.09	0.02	0.09
30	0.99	0.1	0.02	0.09
97	0.99	0.94	0.33	0.93
98	0.98	0.95	0.35	0.94
131	0.82	0.48	0.2	0.5
132	0.83	0.46	0.19	0.48

Table 6.3: Values of the variables involved in a simulation of the model about predator-prey shown in Figure 6.27

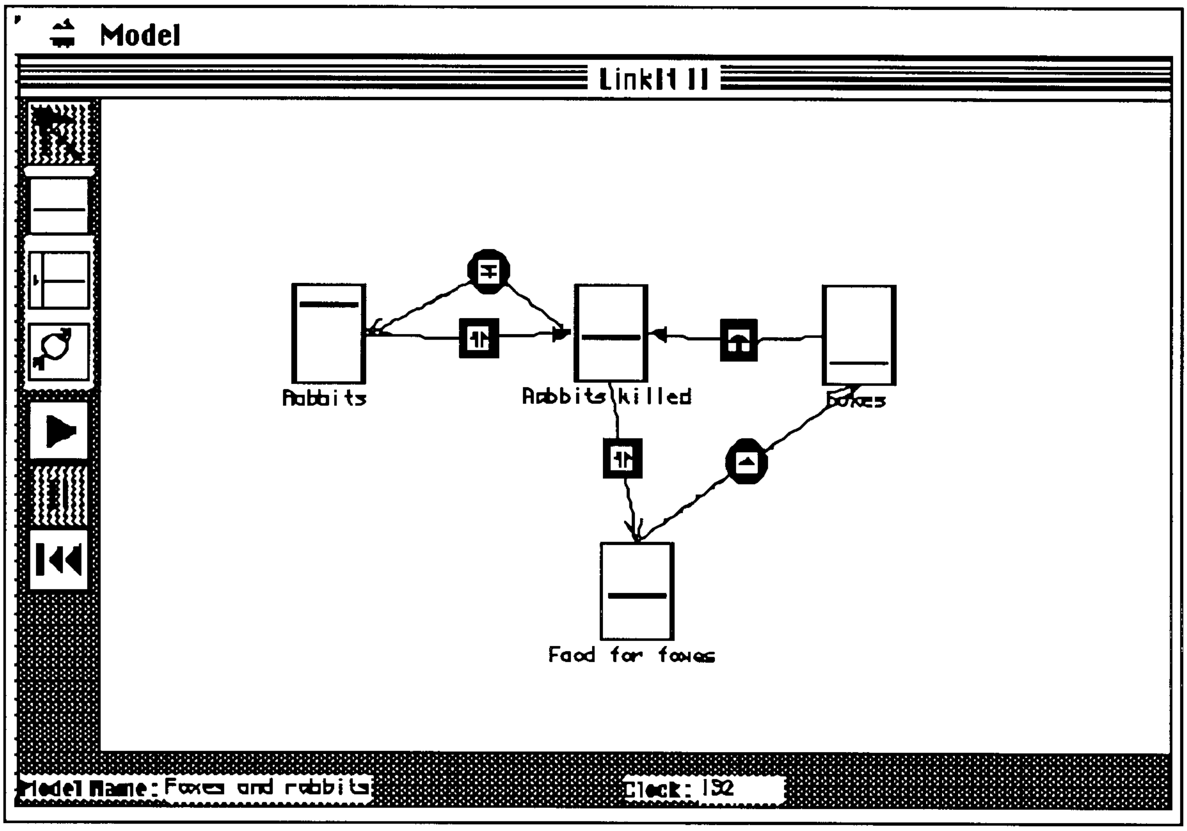


Figure 6.29: Model about predator-prey after 132 iterations

Chapter 7:

RESEARCH QUESTIONS AND METHODOLOGY OF ANALYSIS

7.1 INTRODUCTION

The Core study was intended to extend the Preliminary study where students used LinkIt I. The aim here was not only to pay attention to the newly designed user-system interface and building blocks of LinkIt II (see Chapter 6 for the description of LinkIt II) but also how the students explored and externalised their ideas when using it. In order to achieve this the number of meetings had to be expanded (in comparison to the Preliminary study) and a new set of tasks had to be carefully designed to permit a deeper level of interaction between the students and the software.

7.2 RESEARCH QUESTIONS

The basic research question of the whole thesis, described in Section 3.7, is:

> GIVEN A SEMI-QUANTITATIVE MODELLING TOOL - LINKIT -, DOES IT PERMIT STUDENTS TO EXPRESS AND EXPLORE THEIR KNOWLEDGE IN A CERTAIN DOMAIN (E.G. SCIENCE)?

An important issue related to this question is that any study aiming to investigate the possibilities of using a computational modelling tool in an educational setting pose to the students, at the same time, two demanding tasks:
(1) understanding, use and manipulation of the software;
(2) thinking and learning with the software.

At the time of the Preliminary study, the design aspects of the software and its consequent questions concerning the user-system interaction constituted the main focus of investigation, leading that study to pay much more attention to the first issue mentioned above of understanding, use and manipulation of the software. Nevertheless, the objective remained to give well balanced and careful attention to both issues. Thus, the Core Study can be seen as an investigation of students' successes and failures with the new version of the system in respect to its ease of use and learnability and of how they explored and externalised their ideas when using the software.

The research questions proposed worked more like guiding questions to help the data analysis than a rigorous test of an hypothesis. Given that and bearing in mind the necessity of making simple the process of analysis, as well as presenting it clearly, the guiding questions are organised below along the two levels described above.

7.2.1 Research Questions Concerning Understanding, Use and Manipulation of the Software

As stated above, the questions concerned with (1) were the main focus of the Preliminary study and can be seen as an opening up of the following general subquestion (also presented in Section 3.8):

> HOW CAN SOMEONE'S SEMI-QUANTITATIVE IDEAS BE TRANSLATED INTO COMPUTATIONAL TERMS ?

The analysis in the Preliminary study resulted in the re-design and re-development of the software (see Section 5.2 for the research questions of the

Preliminary study and Section 5.5 for the implications for the re-design of the system) resulting in a new version called LinkIt II (described in Chapter 6). These questions were adapted for this Core study and their answers are mainly presented in Chapter 9.

These research questions are divided into three dimensions. The first dimension contains questions more related to the use of the LinkIt interface. The other two dimensions aim to discuss the use of the main components of the system: variables and links.

Dimension 1: Questions about the LinkIt Interface

1) Is it clear to use and easy to master ?
2) Are the control panel buttons meaningful ?
3) Are the variable shapes meaningful (to represent their properties) ?
4) Are there new features that could be added to the system in order to help to construct models ?
5) Are there new features that could be added to the system in order to facilitate understanding of the models being constructed ?
6) What errors do users commonly make (1- Initially ; 2 - Habitually) ?

Dimension 2: Questions about the Behaviour of the Variables

1) What are the students' ideas of the behaviour of the variables ? Do these ideas change during the process of modelling a problem ? Do they change after using the system ?
2) Do students understand qualitatively the mathematics that controls the behaviour of the variables ?
3) How do students interpret the zero of a variable ?
4) How do students interpret the input combinations of a variable ? What kind of reasons do they have for changing it ?
5) Why do students choose a certain type of variable ? What kind of reasons do they have for changing it ?
6) Why do students choose a certain range for a variable ? What kind of reasons do they have for changing it ?

Dimension 3: Questions about the Behaviour of Links

1) Do students understand how links work ?
2) How do students interpret the properties of a link ? Does this interpretation change during the process of modelling a problem ? Does this interpretation change after using the system ?

7.2.2 Research Questions Concerning Thinking and Learning with LinkIt

This study also attempted to investigate some educational aspects of the software. The questions posed below mainly shed light on aspects of students' thinking and learning when using LInkIt. These questions can also be seen as an opening up of the general subquestion presented in Section 3.8:

> HOW WELL DOES THE LINKIT REPRESENTATION WORK TO SUPPORT THINKING/LEARNING/EXPLORING IDEAS ?

In this study they are also divided along three dimensions which are mainly discussed in Chapter 10.

Dimension 1: Students' Ideas about Models

1) What is the purpose of a model ?
2) What is a good model for the students ?

Dimension 2: Models Created by the Students

1)What is the level of complexity of the models created by the students ?
2) What is the level of appropriateness of the models constructed ?
3) Are there any patterns in the construction of the models ?
4) How did improvement of the models occur ?
    - Nature of change
    - Process of change

Dimension 3: Students Creating Models: What they said and did when interacting with LinkIt

1) Do the students engage in semi-quantitative thinking ?
2) What ideas do the students initially have about a domain ?
    - Do their ideas develop as their work progresses ?
    - Do their ideas become better articulated ?
    - Do they think of new ideas (or change ideas) as their work progresses ?
3) How do the students' ideas relate to the models ?
4) Do the students see connections to other problems ?
5) What relation do the students see between models and reality ?
6) Are the students capable of comparing different models in relation to their underlying structures ?
7) Do the students formulate/test hypotheses ?
    - How do they go about it ?
8) Do the students analyse situations ?

9) Are the students capable of detecting and diagnosing errors ?

7.3 METHODOLOGY

7.3.1 Subjects

The study intended to involve 10 pairs of students but in the end only 9 pairs were able to attend all of the interviews. Among these groups, one of them was not considered in the analysis due to the very low level of participation of its subjects during the interviews.

From the 8 remaining groups of students that were considered in this Core study, 4 of them were aged 13-14 years old and were approaching the end of primary school and the other 4 pairs were aged 16-17 years old and were in the middle of secondary school[1].

The students involved in the study came from typical middle-class schools situated in Rio de Janeiro and Niteroi - Brazil: Colégio de Aplicação da U.F.R.J.; Colégio São Paulo; and Centro Educacional de Niterói.

Each pair of students came from the same school level and both students in each pair agreed beforehand to work with his/her partner in the study.

All students already had some kind of computer experience before using the software. Some of them reported that they had a computer at home and also they were having computer classes as part of their school-curriculum. However, none of them had used a modelling system before.

7.3.2 Procedure and Design

During the design of the Core study some decisions had to be made about the number of meetings and the age of the students.

The Preliminary study gave evidence that the students - although they had used LinkIt I for two sessions only - were able to manage the software and to demonstrate some relevant educational achievements. An important aim of this Core study was to go further in the investigation of the educational potential of the software when used by teenage students. In that way it was necessary to increase the amount of time of student-system interaction in order to permit them to become more comfortable with the system and to use it in problem-situations with different demands. Due to the limitations of time for fieldwork in Brazil, size

---

[1] The Brazilian educational system is based on school levels. The primary school has eight years and a student is expected to finish it when he/she is 14-15 years old. The secondary school has 3 years and the normal student's age in the last year is 17-18 years old.

of the sample and the time-table of the schools from where the students came from, it was only possible to have a maximum of 4 meetings per group with each meeting lasting about 2 hours maximum.

Another important decision involved the age and school level of the students. Knowing that the target users of LinkIt are pre-college (teenage) students who do not (necessarily) have the requisite mathematical knowledge for creating rigorous quantitative models, led to a natural choice of a sample of students distributed in this age interval. However, due mainly to time constraints, it was not possible to work with a representative number of students at each school level inside this age interval (13-18 years old). The decision was made then to work with students aged as close as possible to both extremes of the interval. The choice was to work with groups of 13-14 years old and 16-17 years old[2].

The study was carried out during the months of June to the beginning of July and August to the beginning of October of 1995[3]. The procedure was the same for all studies. The students were grouped in pairs in order to have a better interaction between them, the computer and myself. They spent a total of 7 to 8 hours divided in 4 sessions working with the tasks programmed.

All groups were able to follow the programmed schedule with the exception of Group G which needed 5 meetings to complete the tasks. Among the groups involved in the study, 3 pupils were not able to follow all sessions programmed. In these cases, before the beginning of the next session, I gave a briefing of what had happened in the session they missed.

Table 7.1 summarises the tasks performed by the students and Chapter 8 describes them in detail.

Table 7.2 presents a description of the pairs of students involved and summarises their participation in the meetings.

During the interviews the students were encouraged to think aloud and to explain to each other what they wanted to do in each task. During the first interview, where the Introductory tasks were presented, I intervened more in order to make the students focus on the particulars of the software I wanted to introduce. In the next meetings I tried to keep my interventions as minimal as possible, limiting them to introducing a new activity and/or asking specific questions about what they were trying to do.

---

2 In Brazil the students in the last year of secondary school (17-18 years old) have to follow a special course in order to do exams to gain access to the university. These students are so deeply involved with this course that they simply do not participate on any extra activity that does not have to do with their exams. This is why it was only possible to work with students in the year before (aged 16-17 years old).

3 In Brazil most of the schools have a winter break during the month of July.

Task	Type	No. of activities	Description	Intended session
1	Introductory	1	Describe the experiment and show two examples of models	1
2	Introductory	1	Ask them to write down their ideas about "pollution in big cities"	1
3	Introductory	7	Introduce the main components of the system by showing/constructing some examples grounded in their previous knowledge about the world	1
4	Expressive	1	Construct a model about "migration to big cities"	2
5	Exploratory/ Expressive	3	Explore the model about "predator-prey" Explore the model about "eye-pupil" Explore the model about a "refrigerator". Construct a model about a "heating system"	2/ 3
6	Expressive	1	Construct a model about "diet and healthy life"	3
7	Explor./Expr.	1	Complete some empty models	4
8	Expressive	1	Construct a model about "pollution in big cities"	4

Table 7.1: Summary of the tasks proposed

	Name	Age	Sex	Meetings/ Missing
Group A	Carmen Lucia	13 13	F F	4
Group C	Diego Rogerio	13 13	M M	4
Group D	Lucio Marcio	16 16	M M	4
Group E	Diego Pedro	13 14	M M	4/1
Group G	Andrea Tais	13 13	F F	5
Group H	Fabricio Humberto	17 16	M M	4/2
Group I	Marcio Jair	16 16	M M	4
Group J	Vera Fabricio	16 17	F M	4/1

Table 7.2: Description of the groups, total number of meetings and meetings missed by one of the members of a certain group

All interviews were tape-recorded and the models constructed/explored by them were saved. Also the working sheets used by them and some annotations made during the interviews were kept and used in the analysis.

Apart from Group A, all other groups used a version of LinkIt which automatically recorded in a log file the main system-operations performed by them.

7.3.3 Data Analysis

The data used in the analysis consisted essentially of the transcriptions of the tape recordings of the interviews together with:

- Screen dumps of the models constructed/explored by the students as their work progressed;
- Log files containing the main interface operations carried out by the students;
- Annotations/observations taken during the interviews;
- Work done by the students on working sheets.

The process of data analysis was essentially the same for all data. For each group the available material was examined, aiming to construct a descriptive account of what happened to that specific group when interacting with the system to perform the tasks proposed (each of these descriptive accounts generated a Case study report)[4]. During this process particular attention was paid to the use and learnability of the system and to how pupils explore and externalise their ideas when using it.

After writing the case studies I started noticing that there were no big differences between the older and younger groups in the way they approached the tasks. So when I moved to the next step (where I proceeded with a careful analysis of the case studies), I decided not to take into consideration the differences in their ages and started looking at the case studies as a whole, looking for relevant actions and behaviours of the students in the different tasks. These actions and behaviours were labelled and grouped according to their relevance to the guiding research questions presented in 7.2.1 and 7.2.2. From this set of labels and their "pointers" to the passages in the case study reports and transcriptions, a first version emerged of what was later called a "table of codes". With this first version of the "table of codes", I re-examined the case studies (and in some cases the transcriptions), looking for some actions/behaviours that had been left out of the original and to improve the list of labels in the table. With this improved table of codes in hand, it was possible to start writing an analysis of all the groups.

The decision to present the analysis in a narrative form was mainly because this process provides an effective way to illustrate the sense of discovery and development on the part of the students while working with the software.

---

[4] Appendix D presents a sample of the case studies created here.

Chapter 8:

DESCRIPTION OF THE TASKS

8.1 INTRODUCTION

The Core study consisted of 8 tasks divided between 4 meetings of 2 hours maximum. For each group I tried to follow as closely as possible the tasks described in section 8.2 (see Chapter 7 for Research questions and Methodology of analysis of this study).

The tasks were specially designed to elicit from students their ideas about certain problems when using the modelling tool. The goal was not therefore to see if any significant learning[1] in the domains of the proposed tasks

---

[1] I do follow the opinion of some authors that isonly possible to observe the progress of learning if enough time is allowed for the study.

occurred but to see whether the students could use the system to reason in a relevant way about these domains.

The tasks were arranged in two groups. The first group of tasks (Tasks 1 to 3) had the main purpose of introducing the software to the students. After an introductory part where the purpose of the research was explained and example models were shown, the interface and its components were introduced gradually through a set of 7 activities.

These activities had a broadly common format. The students were presented either with an idea about a certain problem or a simple model, which was used to introduce a new feature of the system. After the presentation of each feature they were asked to suggest or complete a similar model involving the use of that feature.

The second group of tasks (Tasks 4 to 8) were intended to make the students go further in their understanding of the system and at the same time to explore their ideas about modelling with LinkIt. The intention was to investigate the students' behaviour when working with the system in two modes of interaction: students using the software to represent their own ideas about a certain problem (Tasks 4, part of activity 3 of Task 5, 6 and 8) and using the tool to explore the thinking of other people (Task 5 activity 1 and 2). This distinction is called by some researchers *expressive* and *exploratory* modes of learning (Bliss, et al., 1992c) and its importance for the research lies in the different ways in which it is possible to use the system to think about a certain problem.

Task 7, which can be seen as a mix of exploratory/expressive activities, was specially designed to investigate students' perceptions of common underlying structures of models.

Each task is presented in a different section below. The material given to the students, mentioned in the tasks, is shown in Appendix B.

8.2 TASK 1: INTRODUCTION TO THE WORK

In this first Task I briefly introduced myself to the students and explained to them the purpose of the research and what would happen in our meetings. I thanked them for their participation and followed the points summarized below:

- Tell them that the software was developed by me
- Tell them that the idea of the software is to help people to represent problems, situations and ideas in order to think deeply about them
- Tell them that I am going to show them an example of a model already created by other students
    - Load the 'love' model (File: MNamoro. See Figure 8.1 below)

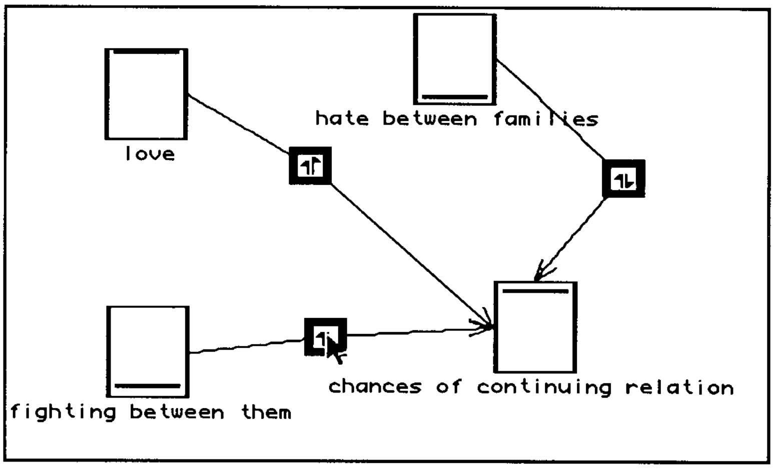


Figure 8.1: First model presented to the students as an example of what can be done with the software

- Tell them what the model is about
- Tell them that the *boxes* (this is the name I used through the experiment as a synonym of *variable*) represent how big or small something is and also some events
- Tell them that the links serve to represent the effect of one box on another
- Tell them about the idea of *same direction* and *opposite direction*
- Tell them that this is not the only way to think about this problem. New boxes and relations can be added
- Run the model and draw their attention to the variables

• Load the 'Nomad population' model (File: MPopNom. See Figure 8.2 below). In this new case, I did the following:

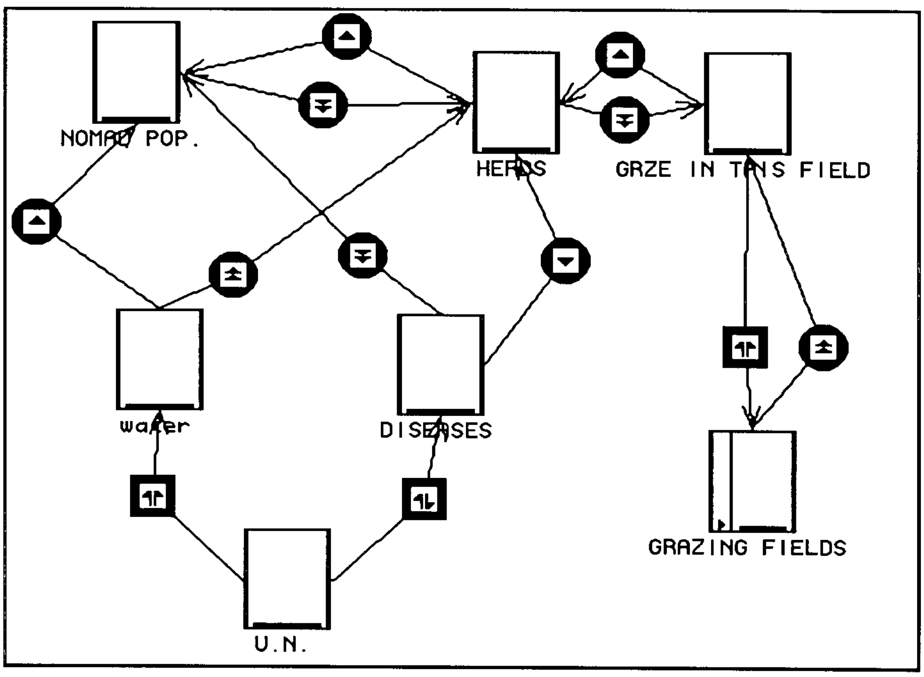


Figure 8.2: Second model presented to the students as an example of what can be done with the software

- Tell them this is a more complex problem which was developed by some students when I was in Brazil last year
- Tell them very quickly the story about the problem and what happened to the nomads when the UN arrived in order to try to help them
- Run the model and draw their attention to what is happening to the levels
- Do not go into a deep discussion about what is happening with the model when it is running (an aim of this second model was to show that the system could represent quite complex problems).

8.3 TASK 2: GETTING TO KNOW STUDENTS' IDEAS ABOUT A PROBLEM

In this Task I asked them to read worksheet 1 (see Appendix B), where a problem about Pollution in big cities is presented, and to answer some questions related to that problem.

The goal here was to give them the opportunity to discuss a problem pervading their daily life and to try to compare their ideas here with the model they would have to construct later, during Task 8.

- Give them the worksheet 1 with the text and questions.

8.4 TASK 3: TRAINING EXAMPLES

The goal of this Task was to introduce the students to the interface and the main elements of the system by constructing and exploring some models with them.

During this Task I took over the system, controlling the mouse and the keyboard. However, if I found that the students were confident about using them, I passed the control to them.

This Task was divided in 7 activities. Activities 1, 3, 5 and 7 introduced new elements of the system. Activities 2, 4 and 6 were used to give the students the opportunity to think further about what was introduced in the previous activities.

8.4.1 Activity 1: Present the Ideas of 'go together' Links, 'direction' of a Link and 'add/subtract'

- Tell them that I can use the system to represent situations like:

- Create the model about the waiter :

Salary --(**GTo, same**)[2] --> *Total salary*

- Run the model with different values of *Salary*
- Tell them that this is not exactly what happens in reality because the waiters also receive tips. So the model has to be something like...

Salary -- (**GTo, same**) ---------->
-- **Add** -> *Total salary*
Tips --(**GTo, same**) ------------>

- Tell them that *tips* has to be added to *salary* in order to know the real value of the salary of the waiter
- Run the model for some different initial values of *salary* and *tips.*
- Show *tips* as zero
- Show *salary* as zero

- Emphasise that *Total salary* only changes when *Salary* or *Tips* change
- Tell them that this model is still not very good because it is not taking into account deductions like income tax and social security. So the model has to be something like

Salary -- (**GTo, same**) ------->
Tips -- (**GTo, same**) --------> -- **Add** -> *Total salary*
Deductions-- (**GTo, oppos**) --->

- Show them how to change the direction of the effect of a link
- Show them how to "read" a model: " If the salary is big then the total salary is big. ....If the deductions are big the total salary will be small"
- Tell them that here in the system a subtraction is seen as an addition with 'opposite direction' sign
- Tell them that in this problem if the *deductions* are big the *Total salary* could be negative.
- The system has a way to represent this idea by changing the size (range) of a box. To do this, the following steps are needed:
    - Open the information box of *Total salary* and change it to any value
    - Run the model using a big value of *Deductions*
- Ask them if they can think of an example similar to this one

---

[2] (GTo, same) means 'go-together same direction' link; (Cum, oppos) means 'cumulative opposite direction' link; 'Add' means combining links with 'addition'; 'Mult' means combining links with 'multiplication'.

• If yes, construct their model and run it
• If no, show them the following example

Time in a day ----------**(GTo, same)** ----->
-- **Add** --> *Time free*
Time spent studying -- (**GTo, oppos**) -->

• Save their model as Ativ1

8.4.2 <u>Activity 2: Explore a Very Simple Model About Road Accidents</u>

The approach of this activity was to present to the students a model about road accidents that did not run very well and try to make them to identify the problems (one problem is about the 'strength' of a link and the other is about how to combine links) and help them to fix them.

The goals here were: (1) to explain/show how a model functions; (2) to show them some important steps when exploring a model; (3) show them the idea of effect of a link; (4) show them the idea of combination of links; (5) to consider whether the model is a good representation of reality; (6) to discuss the introduction of a new variable.

• Tell them that a file with a model already exists in the disk
  • Show them how to load the file MEstra (see Figure 8.3 below)
  • Explain very briefly the idea of the model
  • Run the model with different initial conditions making a parallel between the simulation and real life
  • Ask them to focus their attention on the box *accidents with death* and think whether it represents the real situation well
  • Run the model again if necessary (put *accidents* very high and see what happens to *accidents with death*)
  • Change the effect of the link between *accidents* and *accidents with death* to 'weak' and run the model again
  • Ask them to focus their attention on the box *accidents* and run the model with a big value for *quality of the roads* and zero for *dangerous driving.*
  • Ask them if it represents the real situation well
  • Change the combination of the links to average and run the model again

• Tell them that I want to introduce a new variable called *use of safety belts.* To which variable should it be connected ?
• Should the direction of the new link be 'same' or 'opposite' ?
• What is the effect of the link ?
• How does it combine with the other link ('average' or 'add') ?
• Ask them whether they think this model serves to represent the problem we have on our roads. Ask them to explain their answer
    • If the answer is not a conclusive YES, ask them if they want to change the model to fit it to their ideas

• Save their model as Estra1

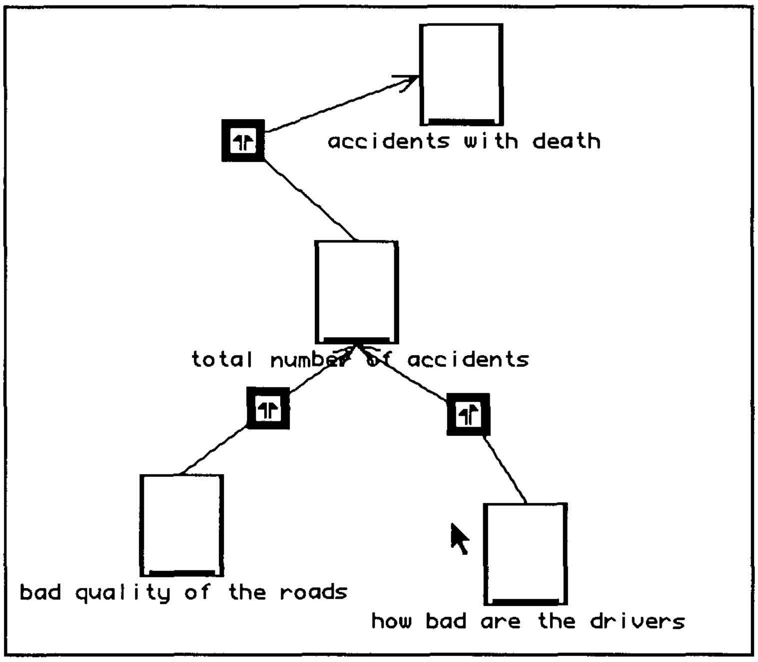


Figure 8.3: Model MEstra

8.4.3 Activity 3: Present the Idea of 'cumulative' Links, 'direction' of a Link and 'add/subtract'

In this activity the students were introduced to the idea of 'cumulative' links and how the direction of this new type of link operates. The approach was very much the same as activity 1 where I constructed a model with the students and brought their attention to what I wanted to introduce.

• Tell them that I can also use LinkIt to represent another type of situation
• Create the following model about the bath

How much the tap is open -- (**Cum, same**) --> *Vol. of water in the bath*

• Run the model for some values of *tap*

- Emphasise that in this case although the *tap* is constant the *volume* is changing
- Tell them that there is another important element in this problem: The drain
- Insert the new variable in the model
- Ask them about the direction of the effect of the *drain* by telling them "The bigger the *drain* is, the faster the *bath* will empty "
- Ask them if they can think of an example similar to this one
  - If yes, construct their model and run it
  - If no, show them the following example

Num. of births per year ----(**Cum, same**) --------->

-- **Add** --> *Population*

Num. of deaths per year -- (**Cum, oppos**) ----->

- Compare these examples with those of Activity 1
- Ask them to pay attention to the different types of representation of the links (a square or a circle)
- Ask them to suggest a name for the kind of problems in the first and second groups
- Save their models as Ativ3

8.4.4 Activity 4: Explore a Very Simple Model About Parking Places

The approach of this activity was to present to the students an incomplete model (only variables without links) about the flow of cars in a parking place and discuss with them the idea in a real situation in order to complete the model with appropriate links.

The goals here were: (1) to explain/show how a model functions; (2) show them again the idea of cumulative links; (3) show them the idea of combination of links.

- Tell them that there is an incomplete model about the flow of cars in a parking place on the disk
- Load the file MEstac (see Figure 8.4)

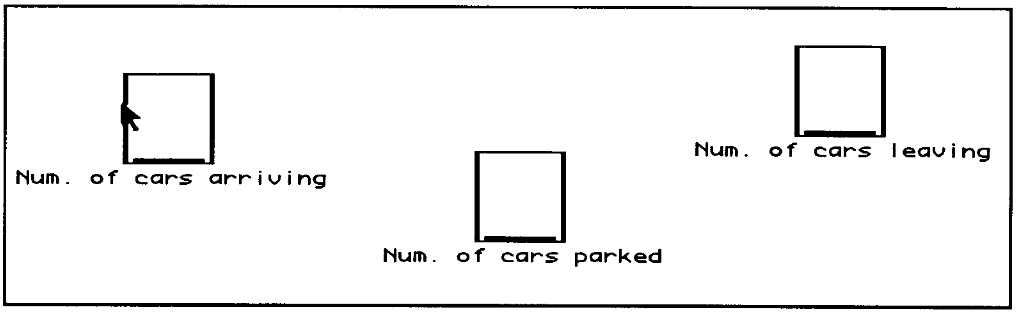


Figure 8.4: Model about parking place (to be completed with the appropriate links)

- Ask them what has to be done in order to observe the flow of cars in the car park
- Ask them about the type of links they want: The one like the waiter problem or the one like the bath problem
- If they don't say anything about the direction, create both links as same direction
- If they give the wrong answer about the type of links, try to make comparisons between this problem and the one about the waiter
- If they give the wrong answer about the starting and ending point of a link, try to draw their attention to what are the causal and dependent factors in the model
- Run the model for different initial values
- Change the sign of the link from *car leaving*
- Run the model again
- Ask them about the combination of the links: 'Add' or 'average' ?
- Ask them to give some suggestions about things that influence the number of cars arriving
- Create the variables and ask them about the relationship

- Save the model as Estac1

8.4.5 Activity 5: Present the Combinations 'multiplication' and 'inverse'

In this activity the students were introduced to the idea of 'multiplication' and 'inverse'. The approach was very much the same as activity 1 and 3 where I constructed a model with the students and drew their attention to what I wanted to introduce.

- Tell them that the third possible case of combination is 'multiplication'
- Construct the following example

Kilos of meat bought -- **(GTo, same)** -->

--- **Add** --> *Total amount paid*

Price per kilo ----- **(GTo, same)** -------->

- Show them, through some simulations, that 'add' does not make sense here (e.g. if price is zero, the total paid is not zero)
- Change it to multiplication
- Run the model with different initial conditions
- Ask them if they can think of an example similar to this one
    - If yes, construct their model and run it
    - If no, show them the following example

No. of customers-------- **(GTo, same)** ---->

-- **Mult** --> *Size of the queue*

No. of tills in the bank -- **(GTo, oppos)** -->

- Run the model for different values of the variables
- Emphasise that the *size of the queue* is inverse to the *Number of tills* (when one is small the other one is big) and we can do this in the system using 'opposite direction' with 'multiplication'

- Tell them that the idea of 'inverse' direction can be achieved by using 'opposite direction' with 'multiplication'
- Show them the special example below

Clouds in the sky -- **(GTo, oppos)** --> **Mult** --> *Sun shining*

- Ask them if they can think of an example similar to this one

- Save their models as Ativ5

8.4.6 Activity 6: Completing Some Models

In this activity the students were presented with a group of three different models that had to be completed. The idea was to give them a chance to experiment with what they had learned about the building blocks of the system and the manipulation of the interface (In this activity I gave the control of the system to them) and tried to help them when necessary.

- Tell them that there is a file called MAtiv6 with some uncompleted models (all links are missing). They have to load the file and complete the models using the appropriate relationships. The models I had in mind were:

Health of the population -- **(GTo, same)----->**
-- **Av-->** *Life expectancy*
Living conditions -- **(GTo, same) ------------->**

Food intake per day --**(Cum, same) ---------->**
-- **Add -->** *Weight*
Energy you spend per day-- **(Cum, oppos) -->**

How economical is the driver --**(GTo, oppos) -->**
--- **Mult-->** *Car fuel consumption*
Distance travelled -----**(GTo, same) ------------->**

• Ask them to try some ideas about the relationships and run the model with some initial values in order to see if they choose the correct relationship. Change the parameters of the relationships and run the model as much as they want.
• Save their models as Ativ6

8.4.7 Activity 7: Present the 'On/Off' Variable

The purpose of this activity was to present the students with the idea of 'On/Off' variables. The approach was to tell them that in some problems we need to represent situations that are not "bigger" or "smaller" but just happen or not. Some examples are about someone beginning to do gymnastics when he/she feels fat or stopping at a petrol station when the tank of the car is almost empty. These two examples were used to introduce the two different possibilities of triggering a variable.

• Construct the following model (*Weight/Overweight* is 'On/Off'):

Food intake --**(Cum, same)**-> *Weight/Overweight*--**(GTo, same)**--> *Gymnastic*
(feedback from *Gymnastic* to *Weight/Overweight*: **(Cum, oppos)**)

• Run the model for different values of *food* and *Overweight*
• Emphasise that when the box highlights it begins to produce an output
• Tell them that the output of an 'On/Off' variable can be of three different kinds (maximum value, minimum value or equal to the level). Try to use the idea of someone doing a lot of gymnastics when he/she feels fat
• Tell them that when the variable is off its output is ZERO

- Ask them if they can think of an example similar to this one
    - If yes, construct their model and run it
- Show them the model in Figure 8.5
    - Change the property 'when it is on' of the 'On/Off' variable by opening the Information box and showing them where this property is presented.
    - Run the model for different values of *Consumption of petrol*

- Save their models as Ativ7

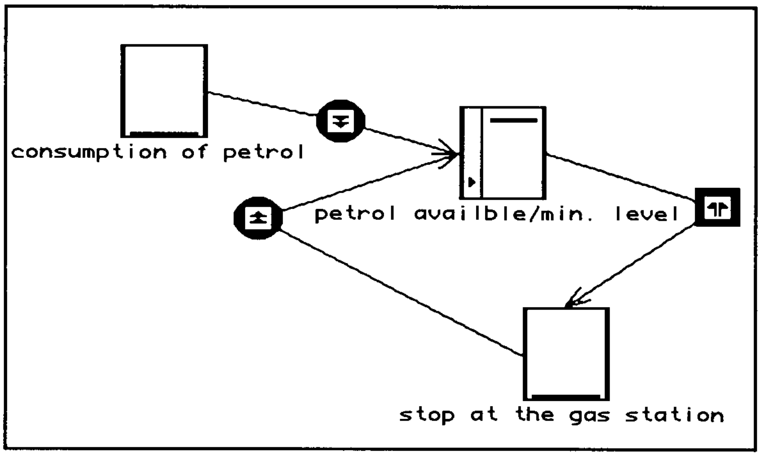


Figure 8.5: Model using an 'On/Off' variable to introduce the idea of 'triggering when below'

This activity was planned to conclude the first session.

8.5 TASK 4: EXPRESSIVE TASK - EXPRESS THEIR IDEAS ABOUT A PROBLEM: MIGRATION TO BIG CITIES

This task was intended to be the first task of the second meeting.

The main goal here was to make them think about a problem and use the system to externalise and discuss their ideas about the problem. The approach was to give them a worksheet where some ideas about the problem are presented (see worksheet 2 - Appendix B) and ask them to create a model that explains why people move to big cities.

8.6 TASK 5: EXPLORATORY TASK - LEARNING A NEW SUBJECT MATTER

This task was divided into three activities, each presenting a new subject matter and asking the students to explore some models provided for them, in order to answer some questions in the worksheets.

8.6.1 Activity 1: Working with the Model About Predator-Prey

The reason for choosing this problem to be presented to the students was the fact that it serves as prototypical example of a category of problems about ecology taught in schools. At the same time, from the point of view of the level of complexity of the model, it is a simple compound of three variables and four links.

The approach to this activity was to give the students an introductory text (see Appendix B - Support text 2) about the subject matter, ask them to load the model already existing in the disk (see Figure 8.6) and ask them to use the model to answer the questions presented in the worksheet 4 (see Appendix B - Worksheet 2). During the process of exploring the model I regularly saved the work they did.

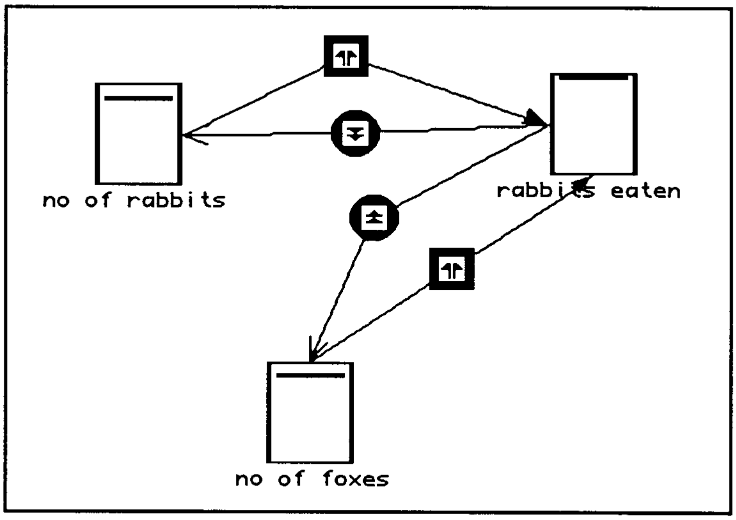


Figure 8.6: Model about predator-prey

It is important to notice that the model shown above does not oscillate (nothing is killing foxes or breeding rabbits). In order to make it oscillate it is necessary either to insert new variables or to set the parameter 'change by itself' so that *rabbits* increase and *foxes* decrease by themselves.

8.6.2 Activity 2: Working with the Model About Eye-Pupil

This activity explored a problem about how the eye-pupil works. Like activity 1 it was a problem discussed in their biology classes at school. However the approach here had two main differences in relation to the first activity . Firstly some questions in the worksheet 3 asked the students to implement some modifications to the model in order to observe a particular event happening (1- someone wearing sunglasses and 2- someone entering and going out of a tunnel). Secondly it asked the students to make some predictions and use the model to demonstrate them.

This model was presented to two groups with minor changes in some links in order to deliberately make the model behave incorrectly. The idea of this modification was to observe how the students react when the model does not work in accordance with their predictions.

The activity began with a text presented to the students about how an eye-pupil works (see Appendix B - Support text 3). After reading the text, the students had to load the model shown in Figure 8.7 (or in Figure 8.8) and answer the questions presented in the worksheet 3 (see Appendix B - Worksheet 3)

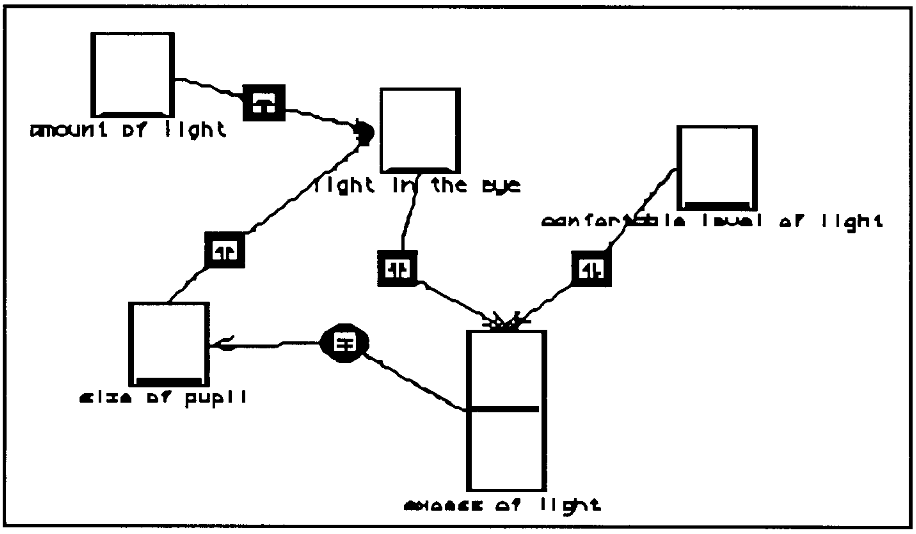


Figure 8.7: Correct model about eye-pupil presented to the students

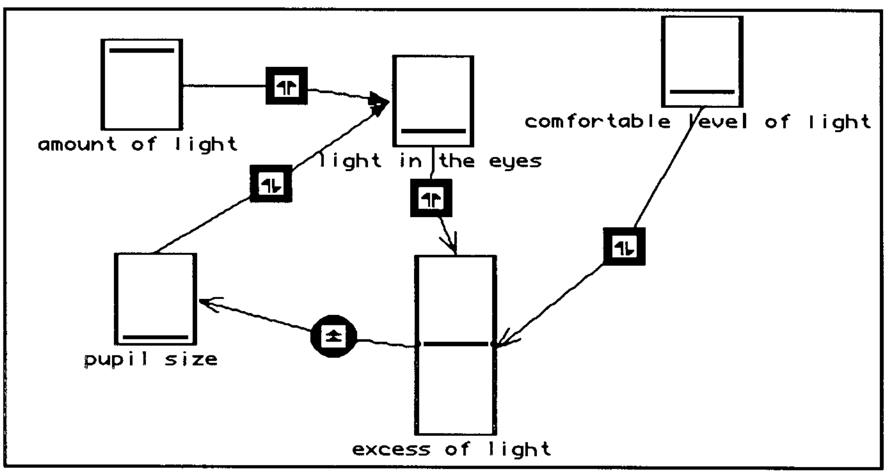


Figure 8.8: Model about eye-pupil with two links changed. Groups E and B used this model

8.6.3 Activity 3: Working with the Model About Refrigerator and Heater

This was the first activity programmed to be presented to the students in the third meeting.

This activity started from the idea that the students were familiar with the functioning of a refrigerator. The main goal here was to see whether there was a change in their explanation about how a refrigerator works before and after using the model. Thus, the approach to this problem was a bit different from the previous two activities. This time, the students did not have a support text to give them some ideas about the problem. Instead, from the very beginning of the activity they were invited to talk about how they thought a refrigerator works.

The second part of this activity asked the students to construct a model to represent a heating system working. The main goal was to observe them using (or not) what they saw in the previous model to construct a new one which, although similar to the previous one, was not part of their daily experience (there are no heating systems in Rio de Janeiro)

- Ask the students to imagine a refrigerator functioning under different conditions such as in a cold and a hot place
- Ask the students: What do you think is going to happen to the refrigerator on a very hot day ?
- Tell them that there is a model about this problem on the disk
- Ask them to load the model MREFRI (see Figure 8.9)
    - Ask them to run the model (The model already has some initial conditions set) and describe what is happening
    - Let them try some simulations and keep asking what they are doing
    - Give them the worksheet 5 and ask them to answer the questions using the model (They have to write down the answer on the sheet. They have to agree on each answer given)
    - Let them try out their prediction and to take note or say aloud what they are testing
    - Ask them to write down their predictions before they test them
    - After they finish, ask them if they found the results they expected
        - If YES, ask them to explain what they found
        - If NO, ask them to try to explain why they did not find it and what has to be done: Change the tests ? Change the predictions ? Change the model ?
    - If they want to make some changes and try things again, let them to do it.

• Tell them that now we are going to imagine a day passing. The temperature is getting higher and higher because of the sun
• Tell them that the system has a way to make a box change by itself and we are going to do this with *sun* to simulate the idea of getting hotter and hotter
  • Set a small value for the rate of increase of sun
  • Run the model again for this new situation
• Ask the students what can be done to measure the quantity of energy consumed by the refrigerator
• Save their models during the process of answering the questions and performing the tests
• Ask the students to construct a model about the automatic heating system of a house

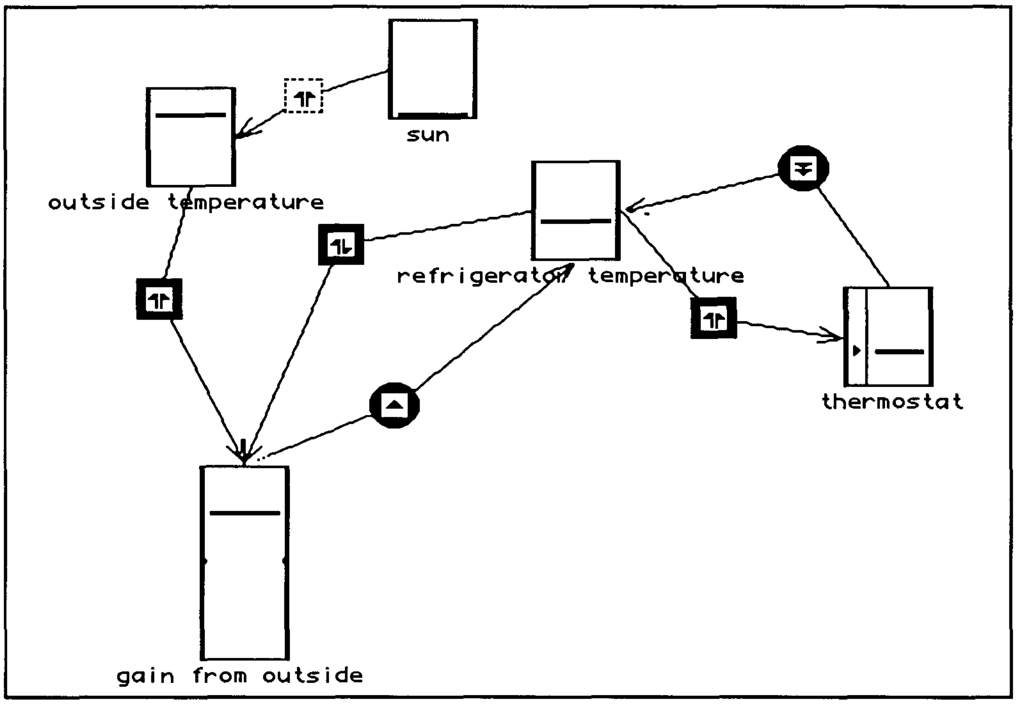


Figure 8.9: Model about a refrigerator working

8.7 TASK 6: EXPRESSIVE TASK - EXPRESS THEIR IDEAS ABOUT A COMPLEX PROBLEM - DIET AND HEALTHY LIFE

The goal here was to see how they express their ideas using LinkIt facilities. A support text (see Appendix B - Support text 4) was given to them.

• There is a model in the disk called MVida that has one variable only: *Your Health*
• Let them discuss the problem freely
• Save their models during the process.

8.8 TASK 7: DISCUSSING ABOUT EMPTY MODELS

This is the first task intended to be presented to the students during the fourth meeting.

The main goal here was to see whether they could go from an abstract model to a concrete one and how complex their models could be when they had a chance to choose the subject.

The approach to this task was to tell them that we were going to play a guessing game. First I presented them with some empty models (boxes without names) and they had to suggest a real problem that fitted it (or fitted with small changes) Afterwards they had a chance to construct a model on their own and give it to me in order to see if I could find a problem that fitted it.

The models presented to them (the names of the variables were erased) were:

Salary --(**GTo, same**) -->
-- **Add** --> *Total salary*
Tips --(**GTo, same**) ----->

Rate of birth -- (**Cum, same**) ---->
------ **Add** --> *Population*
Mortality---- (**Cum, oppos**)------>

Customers -- (**GTo, same**) ---->
-- **Mult** --> *Size of the queue*
Tills ----- (**GTo, oppos**) -------->

Clouds in the sky -- (**GTo, oppos**) -- **Mult** --> *Sun shining*

Food available --(**Cum, same**) --> *Population*
(**Cum, oppos**)

Winter--(**Cum, oppos**) --> *House Temp* --> *Heater* ('On/Off')
(**Cum, same**)

8.9 TASK 8: EXPRESSIVE TASK - CONSTRUCTING A MODEL ABOUT POLLUTION IN BIG CITIES

This was also an expressive task where they could work freely with their ideas about this problem. The goal here was also to see how they express their ideas using LinkIt facilities.

A support text was given to them to "start" thinking about the problem (see Appendix B - Support text 5). The text also contained some questions in order to make them think about how to solve the problem.

Chapter 9:

ANALYSIS OF STUDENTS' UNDERSTANDING, USE AND MANIPULATION OF THE SOFTWARE

9.1 INTRODUCTION

This chapter presents an analysis of the students interacting with Prototype II of LinkIt focusing on the objects (together with their properties) presented in the interface.

9.2 ABOUT UNDERSTANDING, USE AND MANIPULATION OF THE SOFTWARE

A person cannot build a shelter with hammer, nails and timber if he/she does not know how to manipulate these tools and materials. In the same way, to construct models with LinkIt it is necessary to know how to manipulate its interface and its two main computational objects: variables and links.

In this Section we shall see that the students, during the set of tasks presented to them, were able to learn and manipulate the system's interface and all the components it provides. Although they did not at all times perceive the system components in the same way as they were defined, in some situations the students went beyond the implementation of the software, suggesting new features that could make it more powerful and closer to their own ideas.

This Section is divided into three parts. The first part discusses the variables and their properties; the second part is about links and their properties and the third and last part concerns the manipulation of the interface as a whole.

9.2.1 Variables

This Section discusses how the students understood and used LinkIt's variables and their properties to represent their ideas when exploring some domains.

Among other issues, it is interesting to notice that the students often perceived the world as consisting of objects and events and used the system's variables to represent them. This way of viewing the world is not something new as other studies about reasoning with modelling systems (Mellar, et al., 1994; Ogborn, 1992) and commonsense reasoning (Driver, 1983) have indicated.

9.2.1.1 Nature of the Variables

Variables in LinkIt are mainly intended to represent quantities. However, the students also used these 'boxes' with two additional meanings: to represent real-world objects and events.

• <u>Variables as real-world objects</u>

Although the objects represented by the variables were, in most cases, inanimate entities there were cases where they represented a specific person:

> This passage happened with Group A during Task 3 activity 1 when they were suggesting a similar problem to "income of the waiter".
> They thought about representing the performance of a student at school:
>
> F: " (..) *What are the boxes* (variables) *?*"
> C: " *Student*"
> F: " *What does it mean student ? Is it the number of students ?*"
> L: " *No it is the student. If he studies, his grades increase*"

An example of a variable representing an inanimate object is seen in the first passage under the topic below "Variables as objects and events". There *punctured ball* is an object that can be inflated or empty.

• <u>Variables as events</u>

Variables which represented events could be used to represent simple actions such as going shopping or something that controls the behaviour of other variables. Another possibility of this kind was to represent a certain state of something such as a heater being turned on or off.

In most cases where events were used, the students correctly employed 'On/Off' variables to represent them.

• <u>Variables as objects and events</u>

There were also situations where the variable could represent both an event and an object. One example happened with Group E during Task 3 activity 3:

> They were suggesting a similar problem to "Filling a bath tub" and they came up with the idea of inflating a soccer ball that had a puncture (see Figure 9.1). According to the size of the puncture it would be possible to inflate the ball, or not.

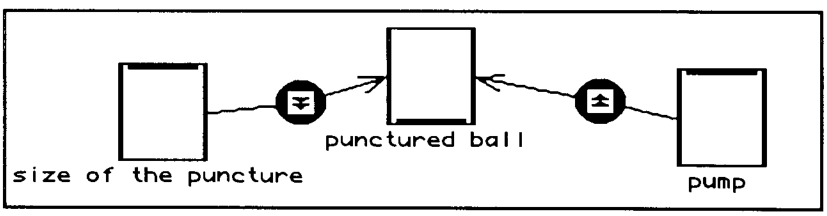


Figure 9.1: Group E: A model about a punctured ball

D: " (...) *the pump would be a square* (variable box) *that would alternate between upwards and downwards... and every time you make it upwards the ball would inflate, when you made it downwards the ball would stay stable* (no change in its volume) *and it would only empty because of this... the puncture* ".

The variable *punctured ball* represents the ball itself and the states of being full (inflated) or empty (deflated).

If we agree that the students, when beginning to model a certain problem, start making a list (on the computer) of the important aspects of the problem to be modelled (see Chapter 10, Section 10.2.3: "Patterns in the Construction of Their Models"), then the variables at the beginning of the modelling process can be seen as a "place" used for collecting a set of interrelated ideas relevant to the problem. In that way in the passage above with Group A the variable *student* could represent not the student himself but the "properties" about the student that are pertinent to the problem: how much he studies, his grades, etc. Actually, later in that activity, the variable in question was opened up into *study, grades* and *attention* . In the same way, Figure 9.1 and the last passage above suggest that the variable *pump* could also be seen as representing how much air has been pumped and *punctured ball* the ball itself and its state of being full or not.

Dependent versus Independent variables

It seems that some students saw no difference between independent and dependent variables. At least at the beginning of their work with LinkIt they behaved in a way that suggested they were not making such a distinction. An example was the situation where they were seeing the relationships as mutual constraint which permitted the reversibility of the links (see Topic "Mutual constraint relationships" in Section 9.2.2.1). In these cases the students permitted themselves to set initial values for all variables, also talking about them as if any of them could be calculated from the others.

As their work with LinkIt progressed they started making a distinction between them, setting initial values to some variables only and seeing them as the causes for other factors. Nevertheless, when there was a feedback loop in the model they were constructing, they continued to see some variables as the independent ones. One such example happened with Group C during the construction of a model about "Migration to big cities" (Task 4):

When they ran the model for the first time (see Figure 9.2) they only set *exodus* and were surprised to see it falling to zero:

F: " *So what were we discussing ?*"
D: " *What determines exodus...*"
(...)
D: " *But I'm not understanding why when...exodus is up there, it goes down when the population is coming.. I'm not understanding this...*"
F: " *What is the relationship between conditions of the people and exodus ?* "
D: " *Oh yes, it is because of this... here the condition is zero...then nobody is coming* (no exodus). "

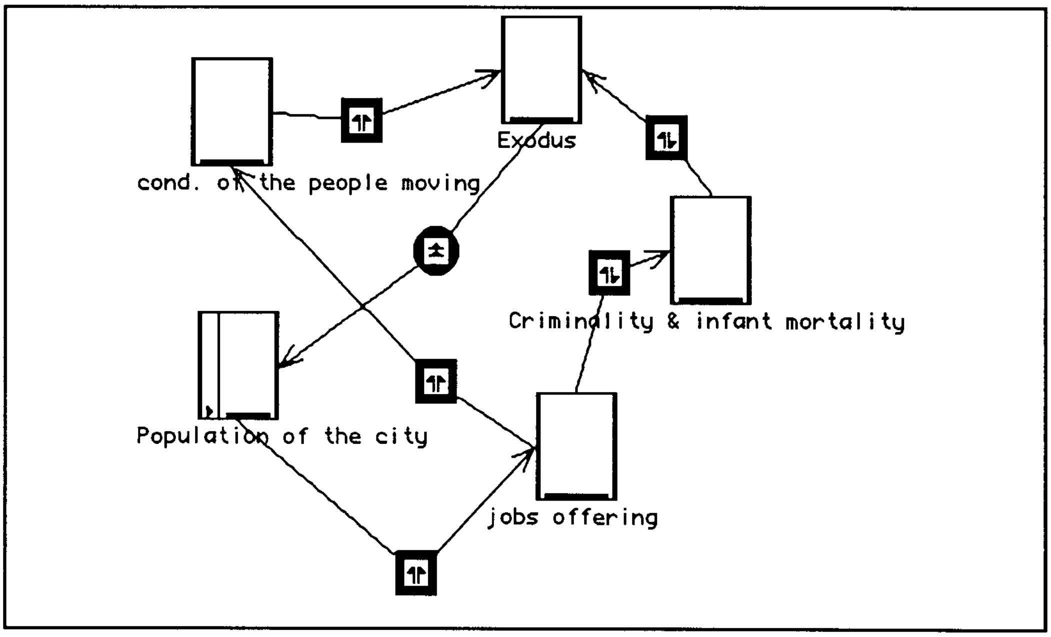


Figure 9.2: Group C: Model about migration to big cities: *Exodus* seems to be an independent variable

It seems that they were thinking about *exodus* as a sort of independent variable (it is the cause of everything) from which everything begins. So *conditions of the people* would only influence *exodus* in the next iteration.

9.2.1.2 Type of the Variables

As we have seen above variables were often used initially to represent objects or events of the world to be modelled. When it was important to represent an event that could happen or not and which could lead to different effects on other variables, the students used the variable type 'On/Off'. A typical example is the model shown in Figure 9.3 where the variable *Total pop. of the city* was made 'On/Off' to represent the idea of bad factors (*lack of homes* and *lack of jobs*) beginning to happen after a certain level of population is reached (see topic "Name of variables" for a discussion about the names used for 'On/Off' variables).

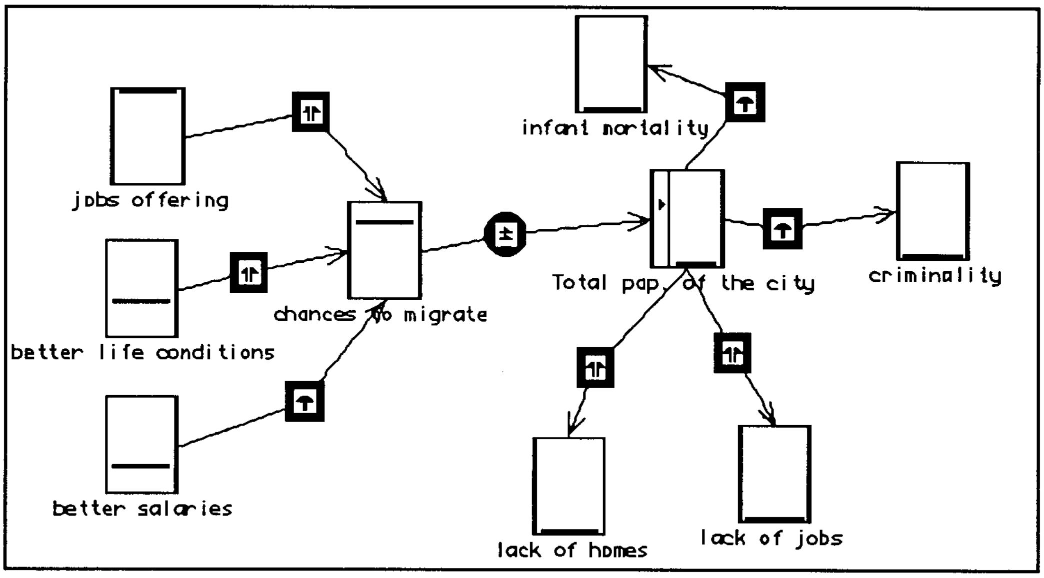


Figure 9.3: Typical use of 'On/Off' variable (model about migration to big cities)

However there were cases of the students using a 'smooth' variable when an 'On/Off' one would be more appropriate.

This passage happened with Group E during the construction of a model about a heating system (Task 5 activity 3).

For one of the students the controlling mechanism of the heater was made by a person. This variable, a bit later, received the name of *turn on* (see Figure 9.4):

D: " (...) *the person turns on the heater, the heater is going up... indoor temperature as well...increasing...When indoor temperature reaches here* (the threshold), *the person turns off the heater, because it is OK inside the house, and here* (heater) *goes down as well*"

...

F: "*Then what is the heater? It has a temperature* (sensor)*inside it, is it ?*"

D: " *No. this box* (variable representing the person) *here is going to change it... it would be the person that turned on the heater. When the person turns it on.... This relation here has to be square* (go together)... *it has to be the same* (...)"

The reason for not using *turn on* as 'On/Off' seems to be the fact that it does not represent an automatic mechanism such as a thermostat but a person responsible for detecting whether the house is hot or cool, who then turns the *heater*[1] off or on. In fact, in this model, *turn on* simply 'follows' the 'On/Off' variable *indoor temp.* and acts as if it were 'On/Off'.

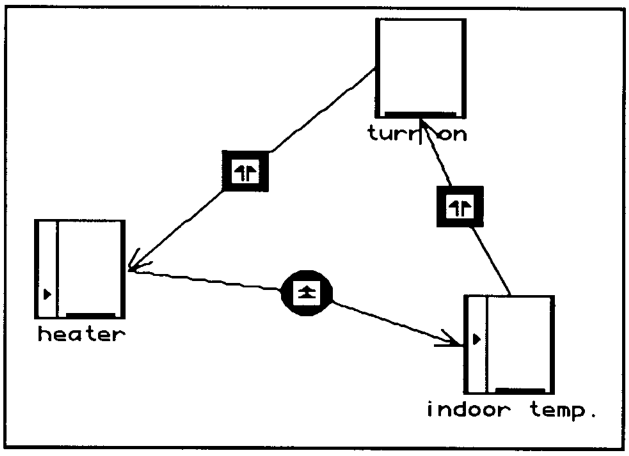


Figure 9.4: Group E: Model about a heating system: *Heater* turns on when above the threshold. *Indoor temp.* turns on when below the threshold

<u>Constructing "modelling gadgets"</u>

An interesting use of 'On/Off' variables was to construct 'modelling gadgets' with the purpose of implementing a certain idea in the model.

Two examples are given below. The first aimed to implement a sequence of events where different things happened. The second was about implementing a constraint imposed by the students on the model.

> This passage happened with Group C during the work with the model about "eye-pupil" (Task 5 activity 2). The students were trying to answer question 2 from the work[redacted]sheet (" How do you show in the model what happens to the eye when someone comes out of a long tunnel on a sunny day ?").
> They first showed the events happening by moving the levels of *amount of light* while the model was running. During this process they kept observing what was happening to the variables (concentrating on *size of the pupil*) and explaining what would happen to the eyes in a real situation.
> I decided to explore further their idea about entering and going out of the tunnel and asked them what could be done to implement it in the model. The suggestion they gave was to create an 'On/Off' variable (later named *tunnel*) connected to *amount of light* via a feedback loop (see Figure 9.5).

---

[1] Another interesting point here was that during our discussion Diego asked me whether it would be possible to make *indoor temperature* with two thresholds. The second threshold was to make the person turn on the heater when the value of *indoor temperature* goes below a certain value.

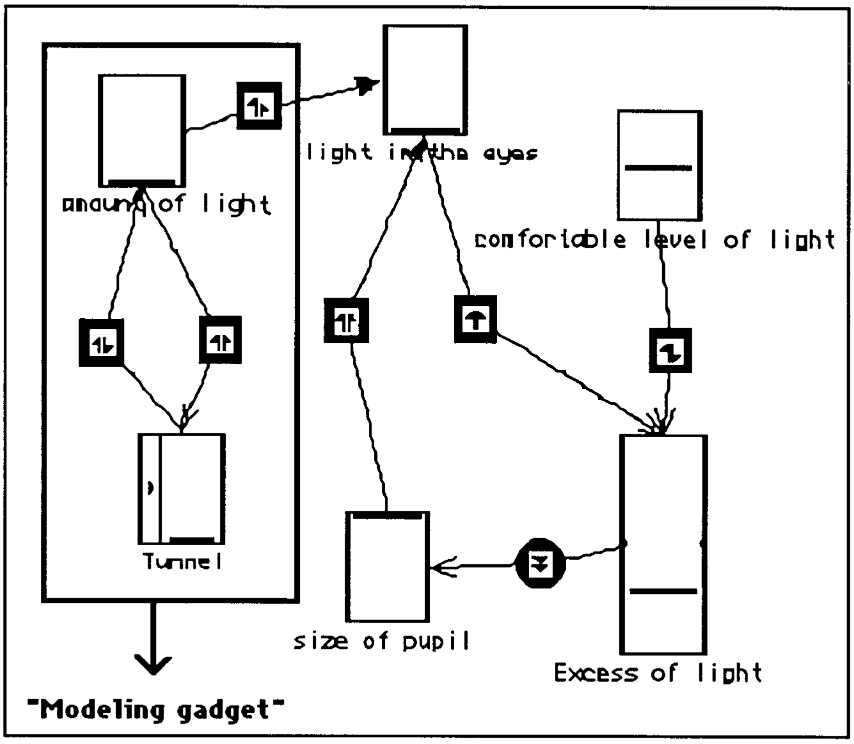


Figure 9.5: Group C: Model about eye-pupil - Using a "modelling gadget" to represent the idea of passing through a tunnel (the variable *Tunnel* is triggered when it is above the threshold)

They explained what they did saying:

D: " (...) *amount of light would begin high because* (the person) *is outside the tunnel, then noname* (tunnel) *would increase* (and) *when it reaches the threshold it would be activated and would diminish amount of light... then amount of light would diminish and the threshold...then the level* (of tunnel) *would go below the threshold and would increase the amount of light because it is the end of the tunnel. The tunnel is very quick* (short). "

The next passage is from Group D, constructing a model about pollution in big cities (Task 8).

According to what they wrote in the work sheet, the solution to the problem would be to diminish the *use of cars* and increase the *use of public transport.* Almost at the end of the task they gave a description of the problem.

F: " *Could you explain to me with this model... in a general way... what you want to represent... your idea about the problem ?* "
L: " *The level of pollution in a city... the use of many cars makes this level increase,* (makes) *pollution increase... but when it arrives at a certain*

level, let's say the government starts a campaign, I don't know, something... to use more public transport... (then) *the people begin to use less cars and more public transport, then the pollution is going to diminish...but the level of cars can't fall very much , otherwise... there is not a city where the people just use buses, where there are no cars... that is why there is an On/Off that doesn't permit use of cars to be very low... when it arrives at a very low level the population begin again to use cars... and there is also the fact that public transport can't be very much because, let's say... anytime you are going to catch a bus it will be full* (of passengers)...*so when it arrives at a very high level of* (use of) *public transport, it* (max. use of buses) *is going to decrease it* (use of public transport)..."

The variables *max. use of buses* and *min. use of cars* (see Figure 9.6) are in the model to implement some constraints of the real world about the use of public transport and cars: " *there is also the fact that public transport can't be very much because* (...)" and "*there is not a city where the people just use buses, where there are no cars* "

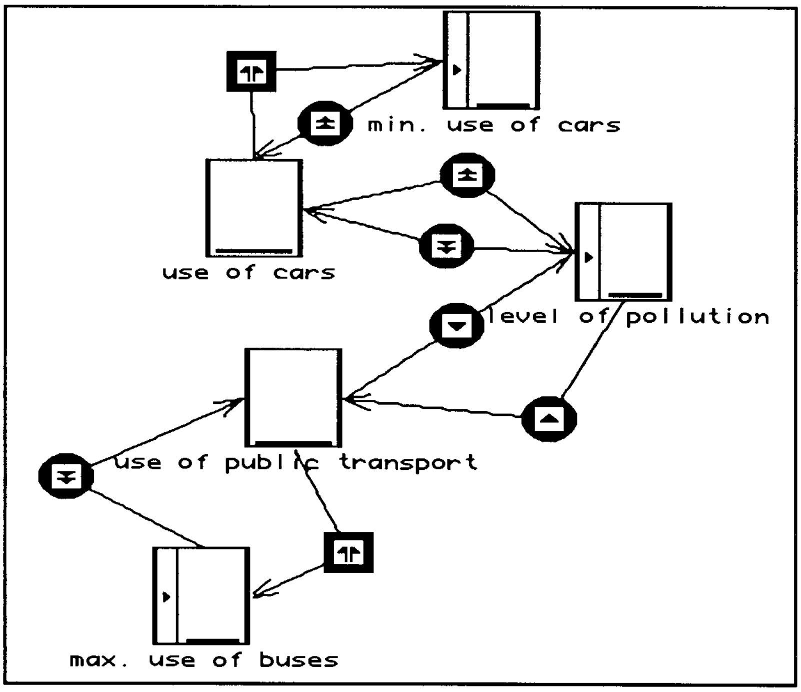


Figure 9.6: Group D: Model about pollution using "modelling gadgets" to implement a constraint on the variable *use of cars* and *use of public transport*

Threshold level

The students did not have problems in understanding the use of the threshold level. They normally associated it with the "critical point" from which something was going to happen.

There was a great preference for making 'On/Off' variables trigger when the amount level was above the threshold. This behaviour can be explained by two facts. First because the concept of an increase causing events/actions to happen when a certain value is reached seems to be a natural way of thinking about events in real life. Second, the students normally wanted a high output when the 'On/Off' variable triggered. However, because the default output combination of the system is set to 'equal to', when the variable was set to be triggered 'when the level is below' the threshold, its output was hardly noticeable. In these cases the students tended to attach the "problem" to the behaviour of the model, to the output links or to a bad choice for the 'On/Off' variable. So, in order to "fix" the problem, they normally changed the link or "recreated" the 'On/Off' variable causing it to be triggered 'when it is above'. Nevertheless, in a real classroom activity this problem could be avoided by drawing the attention of the students to all the properties of the system[2].

<u>Output of 'On/Off' variables</u>

The students seemed to have a preconceived idea that when an 'On/Off' variable was off it still had an output. One indication of this was that they were surprised at the behaviour of a dependent variable in situations where it was connected to an 'On/Off' variable via a 'go together' link. Normally in these situations they did not expect to see the dependent variable going to zero before the 'On/Off' trigger (Under this specific situation it could also be said that the students were interpreting the value zero as nil). One example is shown in the first passage under the topic "Constructing modelling gadgets" above. There Diego said: " (...) *then the level* (of tunnel) *would go below the threshold and would increase the amount of light* (...)". For him when *tunnel* was off it made *amount of light* increase.

Another interesting example of this happened with Group E during the construction of a model about a heating system (Task 5 activity 3). There was an 'On/Off' variable called *indoor temperature* (triggered when 'above the level') which was connected to *turn on* via a 'go together-same direction' link (see Figure 9.4 under the topic "Type of the variables"). After running the model and not seeing *turn on* rise when *indoor temp.* was very low, Diego asked whether it would be possible to make *indoor temperature* with two thresholds. The second threshold was to make the person turn on the heater below a certain value of *indoor temperature* (In other words he wanted an output different from zero when *indoor temperautre* was "off").

---

[2] During the activities I tried to avoid drawing the students' attention to the properties of the 'On/Off' variables in order to see how they coped with this type of variable.

9.2.1.3 Level of the Variables

The level of a variable, which indicates its 'amount', was mostly used to represent semi-quantitative values. So when it was very high it meant "good", "a lot" or "big". At the bottom it represented "bad", "nothing" or "zero".

However, when the students tried to use 'any value' variables they had a tendency to think in a more quantitative way, using absolute numbers to either justify or discard its use. Normally in these cases one of the pair gave a suggestion of using an 'any value' and the other did not accept it because it did not make sense to use negative numbers in the problem. An example happened with Group J during Task 6 (Model about "Diet and healthy diet"):

> The group was discussing how to represent the idea of good and bad health. Fabricio thought that *health* should be made 'any value' where above the middle means good health and below it bad health. Vera did not agree with this idea because it did not make sense to have *health* with negative values. For her the minimum value that *health* could assume was zero therefore it should be an 'only positive values' variable.
>
> Fa: " (...)*it wouldn't be On/Off, it would be that big one* " (referring to the size of 'any value' boxes)
> V: " *No, if you put it like that... if he eats a lot of fat he would have negative health...* (...) *I think it doesn't go below zero. What is worst for health is health doesn't exist... no health. It doesn't exist -3 health, it does exist health 0. Which means health* (equal to) *0 is doesn't have health.*"

Another example happened with Group I during Task 5 activity 3 (working with the model about the "Refrigerator and heater"). In this case it seems that Marcio wanted to represent negative values because a refrigerator, for them, is an appliance that works at very low temperatures, below 0^{o} C :

> Marcio wanted to have the *temperature of the refrigerator* and the *thermostat* represented by the same variable :
>
> M: (pointing to the variable gain from outside - an 'any value' variable) " *I think we needed a box* (variable) *like this here...*"
> F:" *An any value ? Why ?*"
> (...)
> M: " *If the temperature went above zero it would trigger the thermostat* (...) *This box there, temperature of the refrigerator, you would have... the*

trigger arrow would be the thermostat, at temperature X you...(...) *If it goes beyond* (above)*-8º C, it would start the engine* (...)"

9.2.1.4 Range of the Variables

Almost all variables created by the students were 'only positive values' variables (more than 90 %). The task about "Diet and healthy life" produced almost all the 'any value' variables. In this problem they were mostly employed to represent the idea of consumption of a certain group of foods being good or bad for someone's health. When the level was above the middle it meant that that group of food was having a "good" influence on the health of the person and the contrary case represented a "bad" influence. An example is shown in the first passage under the Section 9.2.1.3: "Levels of the variables".

In many cases where 'any value' variables were employed, it was mainly to exploit the possibility of their having an opposite output value when the level went below zero. An example happened with Group E during Task 5 activity 1 (Model about "Predator-prey")

> At the end of the task they decided to represent some ideas that were not in the model such as human beings interfering in the process by cutting trees which would cause a lack of food for some animals. I asked them to implement it and they decided to create a new variable *food of the rabbits* connected to *rabbits.* Although they were seeing the relation between them as 'go together', it was created as 'cumulative' because of another 'cumulative' link arriving at *rabbits.* They ran the model and did not see *rabbits* being exterminated. They then decided to connect *rabbits* to *food of the rabbits* in order to make food diminish and, through the feedback, to make rabbits diminish. After running the model again and not seeing it happening Diego suggested making *food of the rabbits* 'any value' to represent the idea of "*eating less than necessary* (to survive)" (see Figure 9.7).
>
> Pedro did not agree with him because it did not make sense to him to have negative values for *food of the rabbits*. However, Diego convinced him to make the change and try another simulation. This time the model worked in the way they wanted (and Pedro accepted the modification). After that, they started agreeing that the "zero" point of *food of the rabbits* was at the very bottom of the box, the middle was "eating the necessary" (middle = 'normal') and above it was "eating more than enough".

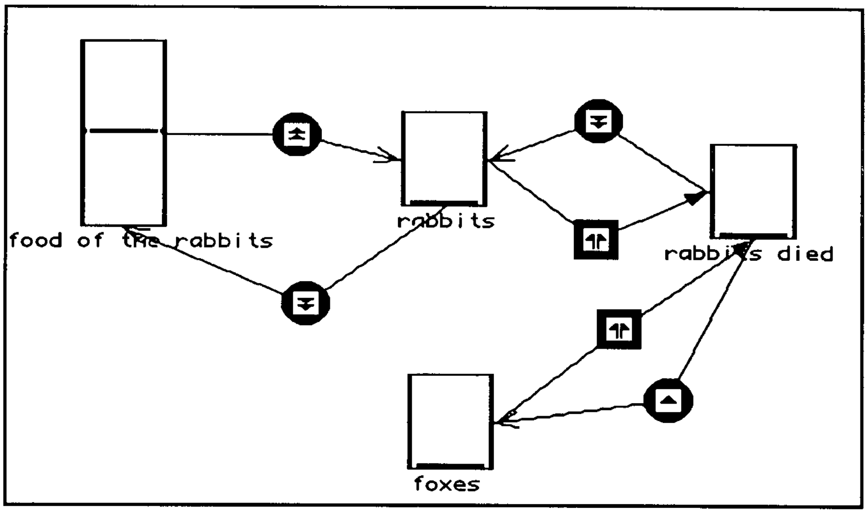


Figure 9.7: Group E: Model about predator-prey: Using an 'any value' variable

There was an interesting situation where a student used an 'any value' variable to represent different moments of time for a certain variable. It happened with Group H during Task 3 activity 4 (Exploring a problem about "Parking places"):

> Just after Fabricio began to do some simulations with the model, he decided to introduce a new variable called *number of cars robbed* (stolen), connecting it via a 'cumulative-opposite direction' link to *number of cars in the parking place*. A bit later, after some more simulations, he came up with an idea about changing *number of cars in the parking place* to 'any value' (see Figure 9.8). Here he wanted to represent the idea of the parking place having some cars before it opened. For him when the level of *number of cars in the parking place* went below the middle it would represent the idea that now the cars that were stolen and/or left the parking places were the ones that "*were already there before it* (the parking place) *opened* . ".

A relevant point about using 'any value' variables was the fact that it gave some students the opportunity to make concrete their ideas about negative numbers. Below there is a passage where the students gave a justification for not using an 'any value' variable:

> This passage happened with Group D during Task 3 activity 1. They were suggesting a similar problem to the one about the "income of the waiter". Their idea was to represent the *retail price* of a product calculated from its *net price* and *income*. After connecting *net price* and *income* to *retail price* (both using 'go together-same direction' links) we discussed:

F: " (...) *In this case does it make sense to have retail price negative ?*"

M: " *No*"

F: " *Why ?*"

M: " *Otherwise you* (the salesman) *are paying for someone to buy it* "

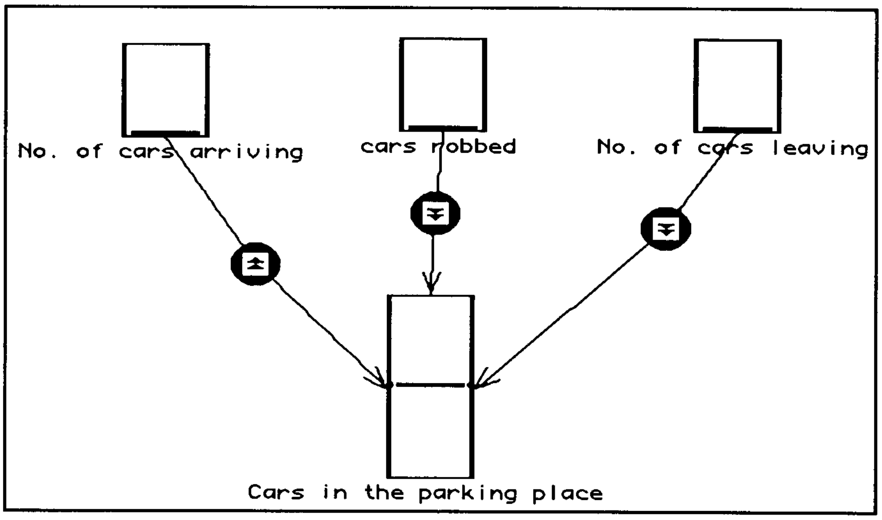


Figure 9.8: Group H: A model about parking place using an 'any value' variable to represent different moments of a certain situation

It seems that the interpretation given to the level and its position in the box has to do with the meaning of the variables being employed and with the desired output for them. Whenever the students wanted to focus on the output of a certain variable, causing it to have a contrary effect on the dependent variable, they tended to use 'any value' variables and interpreted its middle position as a "normal level" where no effect was produced. When the focus was on the variable itself, they tended to use 'only positive value' variables and associated their lowest level to "zero" (absence of something). Sometimes each student of a group insisted on one of these two different points of view. At this time a conflict of arguments about the range of a variable and its associated level emerged and they tried to persuade each other of their ideas (this was exactly the case of the discussion between Vera and Fabricio - first passage in the Section 9.2.1.3 and between Diego and Pedro - first passage in the Section 9.2.1.4).

9.2.1.5 Names of the Variables

The names given to the variables reflected very much what they wanted to represent in their models: often objects and events. They very seldom used words such as "number of ", "total of" or "quantity of" as part of the names they created. It seems that the idea of a variable only came into existence when they associated the object/event represented by the name with the level. The level was there to represent the ideas of total number, quantities, etc.

Sometimes what the students thought about what the variables were representing seemed to have implications for the names given to them. An example happened with Group H during Task 8 when constructing a model about Pollution in a big city. To the similar ideas of "cleaning the streets" and "keeping the sewage system working" (see Figure 9.14) they used the names *street cleaners* and *sewage department*. For the first one they thought about the workers that really do the job. For the second they preferred to mention the governmental department that is responsible for it.

I think that here they are again bringing their experience from the real world to the model: Every day we see many street cleaners working on the beaches and streets of Rio de Janeiro (these people are also part of the folklore of the city being shown in many photographic essays on the city). When they go on strike (which has happened many times in the recent past), the news media and the population make a lot of fuss about it. But when it comes to the sewage workers it is completely different. We do not often see them on the streets, nor do we talk about them every day. So, because they are not seen as people, it is better to represent them as something more abstract.

<u>Naming 'On/Off' variables</u>

Although 'On/Off' variables carry with them two combined ideas (represented by the level and the threshold) the students normally used one of them to give a name to the variable. In most of the cases the chosen name refered to the amount the variable represented, leaving implicit what decide its output. A clear example is the model about Migration to big cities shown in Figure 9.3 above. There the variable *Total pop. of the city* was made 'On/Off' to represent the idea of bad factors (*lack of homes* and *lack of jobs*) beginning to happen after a certain level of population was reached, clearly related to the AMOUNT LEVEL of the variable. Nevertheless, examples where the name is clearly related to the threshold level also occurred. In the model shown in Figure 9.6 above (Topic: Constructing "Modelling gadgets"), the names *max. use of buses* and *min. use of cars* are exactly referencing the threshold level.

It seems there is a pattern to the naming of 'On/Off' variables. When the variable is created as 'Smooth' and later its type is changed to 'On/Off', there is no modification of its name. When a variable is created with the main purpose of controlling a certain event or implementing a constraint, its name is normally related to what decides its output.

Variables changing their meaning

There were cases where the variables changed their meanings to "accommodate" to the way a certain ingoing/outgoing link was working (see discussion and examples in topic "Links and multiple models ideas" under Section 9.2.2.1).

9.2.1.6 State of the Variables ('awake'/ 'asleep')

Although the property of being 'awake' or 'asleep' was not very much used, there was an interesting situation during the construction of the model about "Diet and healthy life" (Task 6) where Group E employed it to see whether a certain idea worked:

> In the middle of the work with this problem, they moved on to the idea of representing a situation where as long as a person was eating a certain amount of a certain product below an established level, his/her health would improve. When this person begins to eat above this level his/her health begins to decline. To implement it they created a new 'normal' variable (*meat*) that would work in conjunction with one of the four groups. In order to try their idea they made a test with just one variable - *meat and dairy products* - putting all other three variables to sleep (see Figure 9.9).
>
> D: " *All are sleeping now...only meat and dairy products now...now opposite here...Here will increase*"
>
> The model is running...
>
> D: " *When it arrives up there, health has to decrease a lot...*"
> R: " *Now it is up there...*"
> D: " *It worked... Now we have to do* (the same) *with all* (variables)*...it is going to be a bit complicated...*"

Some students were not sure what happens to the links when a variable is sleeping. A possible reason for this confusion may be the fact that when a variable is sleeping only the contour of its box becomes dashed, leaving an uncertainty about the links. An example happened with Group D during Task 5 activity 3 (Model about "Refrigerator and heater"):

We were discussing the variable *gain from outside*. The students were suggesting a possible way of not using it in the model. So they decided to make it 'asleep' and create a bypass from *outside temperature* to *temperature of the refrigerator.*

F: " *Do you want to test it ?* "
L: " *Delete it* (gain from outside) *or put it to sleep ?* "
M: " *Put it to sleep... if necessary we ...*"
L: " *Is it enough to make it only with the box or* (you) *have to make it with the links ?*"

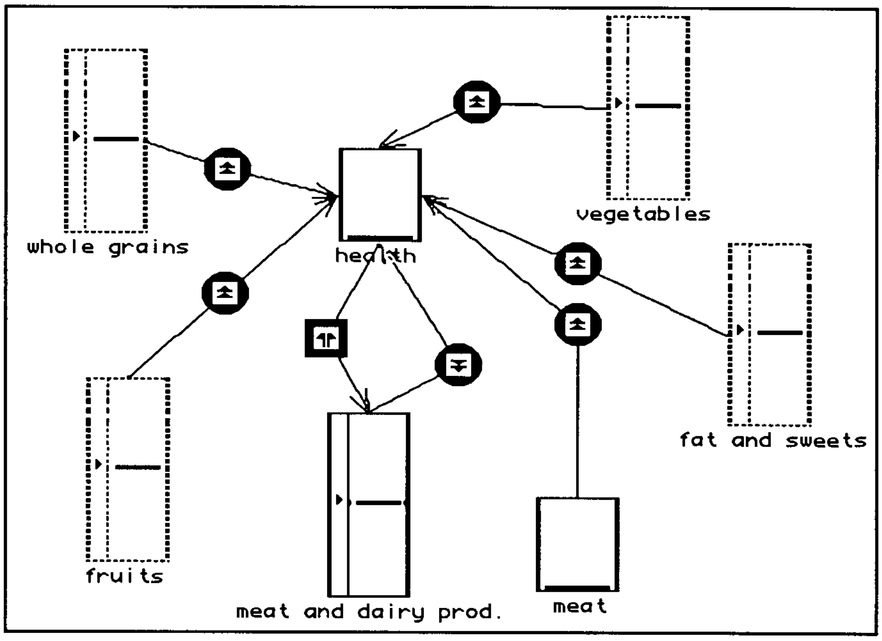


Figure 9.9: Group E: Model about diet and healthy life. Making variables 'asleep' to try out an idea

9.2.1.7 Independent Variation of the Variables ('Change by Itself')

This property was introduced during Task 5 activity 3. The students understood the idea and used it to make the variable *sun* change its value to represent the idea of a day passing.

One of the rare examples of using this property outside the task where it was presented, happened with Group I during Task 6 (Constructing a model about "Diet and healthy life"):

After they finished their model (see Figure 9.10) I decided to ask them about the purpose of their model in relation to time passing:

F: " *This model there, does it represent an increase in the life span of a person... how are you seeing it ? Time passing and then that is what happens...*"

M: " *Time passing... isn't it interesting ?... You did not demonstrate it very well for us... we saw it in a general way... this feature about change by itself...As time is passing we could see how it* (the model or the person represented by the model) *reacts* "

After that they used the property 'change by itself' to make the variable *age* increase with time.

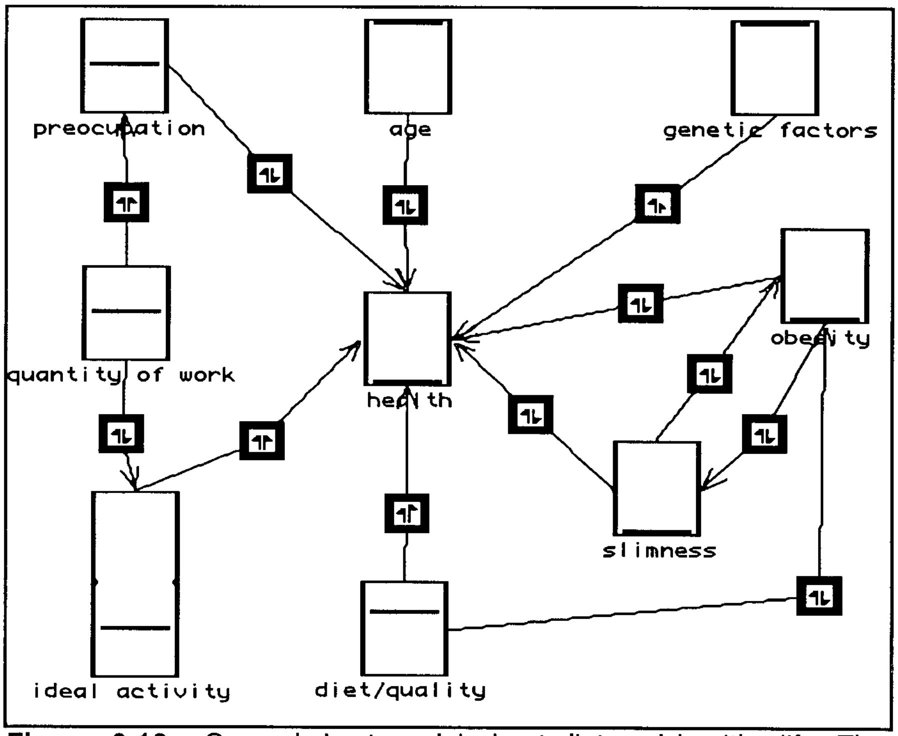


Figure 9.10: Group I: Last model about diet and healthy life. The variable *age* is set to increase

9.2.1.8 Using Input Combinations

The majority of the links created by the students were combined using 'add'. In fact since the beginning of the tasks they talked about the variables causing a certain effect on a dependent one in terms of values being added/subtracted.

<u>Use of 'average'</u>

Some students tended to interpret 'average' as a way "to force" the result to stay between the inputs:

Group A was working with the model about Diet and healthy life (Task 6) and after the second simulation they did with their model, they saw the variable *health* going to zero. Then Carmen noticed it was doing 'add' and they decided to change it to 'average':

L: "*No. It has to be average* "
F: " *Average ?* "
C: " *Yes. I think it is average* "
F: " *Why average ?* "
L: " *Because there is something making it* (health) *bad... fat and sweets but because he eats a lot of fruits and meat* (these are healthy food for them) *it stays in a balance. Like in a scale with what is not good* (in one side*) and with what is good* (in the other side)."

I think they wanted 'average' here because they were thinking about things that were counterbalancing each other. Another relevant aspect that may contribute to this misconception is that in Portuguese the words 'balance' and 'average' are the same ("media").

Another justification given for the use of 'average' was that in the problem they were focusing on, the values were not very "precise" (they were approximated values):

The following passages happened with Group D during Task 3 activity 6 (Completing some models):

During the work with the model about fuel consumption they said:
M: " *I think it is average. Do you agree with average ?* "
L: " *You do not have a way to measure it... in this case I think it is average...*"

They more or less repeated this behaviour when working with the model about life expectancy:
F: " *Why do you think it should be average* (between life conditions and health) ? "
M: " *Well... to add up two abstract values like these, you do not have a way to do it... it should be something more general...*"

This idea of working with inexact numbers also happened with Group H during Task 3 activity 2 (Exploring a model about "Accidents on the road"):

> Fabricio did not like to see *accidents* very high when *bad quality of the roads* was very high and *dangerous driver* very low. I changed the combination of the links to 'average' and after running the model he said:
>
> Fa: " *Now it is more presentable than the other* (using 'add') *because there, the number of accidents with death is very big* (...) *it could have a high number of accidents on the road but not with death...*(...) *this one is representing the idea better* "
> F: " (...) *is it possible for you here to give an explanation why using 'average' here in this example is more interesting than doing 'add' ?* "
> Fa: " *It became more presentable... and also we are not dealing with exact numbers.* "

<u>Use of 'multiply'</u>

Although it seems they understood my introduction to the idea of "multiplication" through the use of prototype examples, in most of the cases it was very difficult for them to construct a similar example. Even during later tasks they just used it as a "last resort" to try to fix a problem in their models.

Nevertheless the idea of "inverse" - which could be achieved by using 'multiply' combined with 'opposite' links - was very much employed by the students though in a "wrong place": They tended to see an 'opposite' link as a way to make a relationship work as "inverse" (no matter what input combination was set). Some examples of this case are shown under the topic "Using opposite as inverse" in Section 9.2.2.4.

9.2.2 <u>Links</u>

9.2.2.1 Giving Different Meanings to the Links

Links were implemented in LinkIt to represent relations between variables. Also according to its conception, the direction of the arrow on the screen represents the direction of the effect of one variable on another. However we shall see in this Section that these were not the only ways the students understood and employed them.

<u>Cause-effect and semi-quantitative reasoning</u>

Links were mostly used to represent a causal relation between variables. Normally when they talked about them they also included a description of the direction of cause of the relation using semi-quantitative reasoning.

This extract happened with Group D during Task 3 activity 1 while creating the model about the "income of the waiter".

F: " (...) *So how is this relation between bills to pay and total income ?*"
L: " *Bigger the bills, smaller the income.*"

The same group in the same activity when creating the links in the problem about "time in a day" suggested the creation of a link between *time studying* and *free time* saying:

M: " *Bigger time studying, smaller the free time*"

In the next passage Group H was constructing a model about "Migration to big cities" (Task 4) and Humberto was justifying the type of a link between *jobs offering* and *life conditions*:

H: " (...) *while jobs offering is diminishing the life conditions of someone living in the city... the life conditions in the city is diminishing as well, (it) diminishes together* (with *jobs offering*)."

Values Flowing

However, there were also examples where the students used links to represent the idea of something flowing from one place to another. They did this in two main situations. One involved at least three variables (1 independent and 2 dependent) and represented the idea of dividing something and giving its parts to someone (like "slicing a pizza and sharing it"). The passage below gives an example of this behaviour:

This model was about "time in a day" (Group A -Task 3 activity 1). It had 3 variables: *time in a day*, *time spent studying* and *free time*. I asked them how we could connect the variables and they suggested creating two links from *time in a day*, one going to *time spent studying* and the other to *free time*. Both 'go together-same direction' with 'normal' weight. They also suggested a third link connecting *time spent studying* and *free time* with 'opposite direction'.

L: "(...) *time spent studying and free time are equal to time in a day . The sum of the two is equal to time in a day. Like I have one hour free* (time in a day) *and I take 40 minutes to study* (time spent studying) *and 20 minutes to walk around* (free time)"

The comment " *Like I have one hour free and I take 40 minutes to study and 20 minutes to walk around* ." suggests that *time in a day* can be sliced (or divided) in two parts. One part goes to *time spent studying* and the other to *free time*.

The second possibility normally involved two variables and represented something flowing from one place to another and while it was happening the causal factor diminished and the dependent factor increased (like a "tank of water emptying"). One example of this case is shown in the passage below:

> This extract happened with Group J during Task 5 activity 3. I had asked Fabricio how he would represent the consumption of energy by the refrigerator and he suggested the creation of another variable - *usage of energy* - which he wanted to connect to *thermostat* (with a link from *usage of energy* to *thermostat*).
>
> F: " *If I'm thinking about consumption of energy, how much energy the refrigerator consumes, I want to measure...*"
> Fa: " (...) (the link) *could be from usage of energy to thermostat... if it was to calculate the energy that is being liberated* (in order) *to* (keep) *the refrigerator working ... But now if it was the case of consumption* (of energy) *it* (the focus of the problem) *could be this relation from thermostat to consumption....* (in this case) *We are going to have an indication of how much it* (the refrigerator) *is consuming to keep it working.* "

He was seeing *usage of energy* as a "tank" of energy. So when the *thermostat* turned on the tank of energy would empty ("*liberate energy* ") to keep the refrigerator working and *consumption of energy* would increase.

Mutual constraint relationships

Also during the beginning of the work with LinkIt some students interpreted the links as a "mutual constraint" relationship. As a consequence of this interpretation the students, in some cases, did not make a distinction between dependent and independent variables. A special case of this situation was to envisage the "reversibility of the links", where it would be possible to calculate the value of the independent variable from the value of its dependent factors. In the first example under the above topic "Values Flowing", what they said - "*The sum of the two is equal to time in a day* " - suggests that *time in a* *day* (the independent variable) could be also calculated from its dependent variables.

Although this behaviour was quite rare it suggests an interesting way of interpreting links which could be related to the algebra taught at school. From there it is learned that in an equation such as P * V = n * R * T (universal law of gases), any of the four variables can be determined when the other three are known. In the same way for them once they know the values of *time spent studying* and *free time,* the value of *time in a day* can be automatically calculated. If LinkIt worked like that more types of problems could be modelled by the system. However the computational aspects of implementing such an idea are not so simple.

Links and multiple models ideas

Another interesting aspect of using links was the interrelation between them and the main idea(s) governing the construction of the model. In the example below, two different ideas in the students' minds, led them to make changes in the links and variables used in their model.

This passage happened with Group E during Task 7 when they were trying to find a problem that fitted in the following structure:

XXX('Smooth') -- (**Cum, same**)---> YYY('Smooth')

(Cum, oppos)

Their initial idea was about someone having a certain amount of money and going shopping. When he/she begins to buy things his/her money begins to go down:

D: "*Let's see if it could be like that... bigger the money...bigger are the shopping...*"

P: " *Bigger the shopping ... smaller the money...*"

F: " *If there is a lot of money, shopping increases and increases and increases... doesn't it ?*"

P: " *It is going to do a lot of shopping* "

Up to this point *money* could be translated as how much time someone spends shopping which can also be seen as how many things you buy.

The idea sounded good but when they ran the model they expected *shopping* go down to zero (after *money* finished), giving another justification for that:

D: (the model is running): " *Bigger the shopping... smaller the money...*"
P: " *Money finished.*"
F: " (...) *money finished... does shopping stay up there ?* "
D: " *No... but...*"
P: " *Yeah... it has to stop shopping...when money finished, it had to stop shopping...*"
D: " *Shopping would diminish.* "

At this point *shopping* is more like an event that could happen or not.

I then decided to let them "fix" the model, changing whatever they wanted. They first thought about making *money* you have as an 'On/Off' which would turn on when the level goes 'below the threshold'. When it turned on it would make *shopping* go to zero (see Figure 9.11)

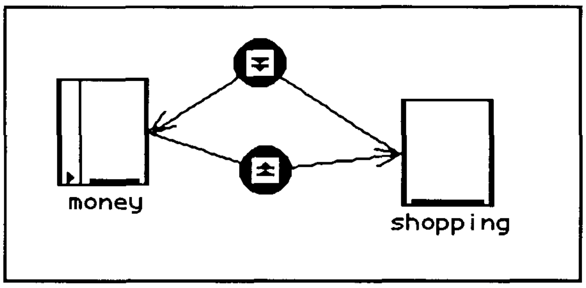


Figure 9.11: Group E: Model about money and shopping

But when they ran the model with level of *money you have* above the threshold, they did not notice any change in the variables. The real reason for this is the use of cumulative links and the fact that when *money you have* is "off" it makes its output equal to zero. However, they decided to make *money you have* 'smooth' again and *shopping* 'On/Off' triggered when above the threshold (Here they simply transferred the problem in their model from one variable to another). Again the model did not work. I then decided to make them think harder about the way they were using the 'On/Off' variables and in the end they realised that they should change both links between *money you have* and *shopping* to 'go together'. They ran the model again and became satisfied saying:

D: " *It is correct...*"

P: "*Because when...*"

D: " *Because when he has money he buys. When money finishes he stop buying.*"

It is clear from the passages above that they were moving between two different ideas to be modelled. One about "having money and going shopping" (talking about an event that happens or not) and another one about "while someone has money he/she goes shopping and buys different things" (talking about accumulating things). This is a very important distinction that should be made in advance. However, it is also a big leap in the process of modelling which even experienced modellers do not always make. If it was a real school activity the teacher, together with the system, could lead them to think further about these different ideas.

9.2.2.2 Choosing a Link

The choice of a link to connect two variables is mainly based on two factors: on the expected behaviour of the dependent variable on the screen and on the relation of the model with time. So, 'go together' links could be used to produce a quick effect on the dependent variable (a big change in its value) and/or used in "snapshot events" where time was not an important variable. In their turn, 'cumulative' links were mostly used when a smooth variation of the dependent variable was desired and/or to represent situations that had to do with time passing. Below some examples of these cases are presented.

This extract happened with Group H during Task 5 activity 3 when they were creating a model about the heating system. They first created a model with three variables: *cold outside*, *cold inside the house* ('On/Off') and *heater* ('On/Off') and connected *cold outside* to *cold inside the house* ('go together-same direction' link), *cold inside the house* to heater ('go together-same direction' link) and *heater* to *cold inside the house* ('go together-opposite direction' link). The idea was that when *heater* turned on, it would make the *cold inside the house* diminish. They then ran the model and commented:

Fa: " *I never saw a cold* (talking about cold inside the house) *diminish so quick*"

H: " *That is why it should be gradual ...*"

Here the expected behaviour of the variables on the screen dictates the choice of the link and, for them, a quick effect is achieved by using 'go together' links whereas 'gradual' (a nickname for 'cumulative' links) is for things that change slowly. In the same way, Group G during Task 3 activity 4 explained both types of link saying:

> Tais was confused about the ideas of 'cumulative' and 'go together' links. I asked Andrea to explain to her when one or the other should be used:
> F: "*So, Andrea explain to her what do you understand when it is square* ('go together' link) *or ball* ('cumulative' link)"
> A: " *When it is square... the level goes up and down* (meaning something quick)... *when it is ball, it goes up following... it goes up slowly...*"

The following passage happened with Group C during Task 3 activity 4 when they were completing a model about parking place (there were only variables on the screen. The links were missing). They justified the use of 'cumulative' links saying:

> R: " *Now it is not square* ('go together'). *It is ball* ('cumulative') "
> F: " *Why is it ball ?* "
> D: " *Because the more cars arrive, the bigger will be the number of cars in the parking place*"
> F: " *Yes but the more income I have, the richer I am, the bigger is my total income..in that case of the waiter it was square and more income he had...*"
> (I was recalling the previous model about the "income of the waiter" where the links were 'go together')
> D: " *Yes but there...with the waiter it only counted at the end of the month* "

Here they were observing a phenomenon over time, and therefore the links were 'cumulative'. In the previous model (about the "income of the waiter"), the justification for using 'go together' links was that they were interested in a "snapshot" of the situation where time passing was not taken into account ("*it only counted at the end of the month* ").

In the passage below the idea of time passing is explicitly used as a justification for using 'cumulative' links:

> This passage happened with Group H during Task 3 activity 3. Fabricio had suggested a model about using a stock of energy where consumption of energy makes existing energy diminish. He was explaining why he used a 'cumulative opposite-direction' link .

Fa: "*With time it* (stock of energy) *falls... and falls...*"

F: *If I increase the consumption of energy what happens to existing energy ?* "

Fa: " *It will go down quicker*"[3]

It is important to say that it is not being claimed that the students did not reason about the relations in the same way they were specified during the implementation of the system. Quite the contrary, the passages above suggest that their ideas were in accordance with the purpose of 'cumulative' and 'go together' links though their explanations were very much qualitative and strongly based on what they perceived from the visual feedback given by the system. For instance when Fabricio said "*With time it* (stock of energy) *falls... and falls...*", he correctly perceived that as time goes, the stock of energy decreases (de-accumulates) because an electrical appliance is consuming it.

Nevertheless, there were situations where they talked more explicitly about the mathematical ideas of 'cumulative' and 'go together' links.

In this passage Group D was working with the model about eye-pupil (Task 5 activity 2). There was a 'go together-opposite direction' link between *size of the pupil* and *light in the eyes* and they changed it to 'same direction':

F: " *You changed the link there, between size of pupil and light in the eye, didn't you?*"

M: "*Yeah*"

F: " *You set it to same direction. What was the argument you used to...?*"

L: " *Bigger the size of pupil, more light goes through... then light in the eyes has to be directly proportional to the size of pupil* (...) "

Here the idea of 'go together' is interpreted as "directly proportional" which is exactly how this type of link is mathematically interpreted by the system.

The next example is about using 'cumulative' links. An interesting point to observe is that Fabricio not only talked about money being accumulated but he also made a distinction between this situation and the first model we worked with (Task 3 activity 1).

---

[3] This passage can also be seen as an example where Fabricio is qualitatively considering *consumption of energy* as a rate of change of *energy*. According to him when the rate of change increases, the value of *energy* will go down quicker .

This passage happened during Task 7 where they had to find a problem that fitted on a model that was on the screen. The structure of the model presented was:

XXX ---- (**Cum, same**) ---->

----- **Add** --> ZZZ

YYY -- (**Cum, oppos**) ----->

After doing some simulations with the model Fabricio said "*it is easy*" and suggested the model shown in Figure 9.12. The explanation he gave was:

Fa: " *There are lots of bills. Salary is not enough to pay them, then your income of the month is going to drop, drop, drop....*" (*money* keeps dropping)

His example was very similar to the first model he worked with in Task 3 activity 1. The only difference was that there the links were 'go together'. I then decided to argue with him about the links in order to understand more clearly his interpretation for the 'cumulative' links:

F: " (...) *If I earn $ 1000. every month and spend $ 200. my income will be always equal to $ 800. ...*"
Fa: " *Your income is going to increase every month $ 800, $ 800, ...*"

After this discussion he promptly changed the name *income of the month* to *income.*

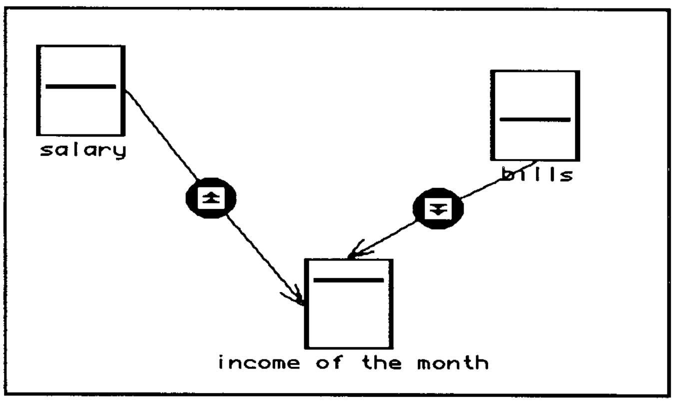


Figure 9.12: Model suggested by Fabricio using 'cumulative' links

It is clear that he was seeing the situation as something that was linked with time and money (*income*) accumulated/de-accumulated during this period.

'Go together' links and initial values

There were some situations where the students wanted to set an initial value for a dependent variable which had an ingoing 'go together' link. It seems they were seeing it as a special case of 'cumulative' links where the simulation ran only once.

This passage happened with Group H during Task 6 when constructing a model about "Diet and healthy life". They first constructed the model shown below (Figure 9.13). The causes of diet were made 'On/Off' to represent the idea that above a certain level that component of the diet will begin to be harmful.

When they began to test it, they set an initial value to *diet* to represent the idea of someone having a good quality diet. The independent variables were left zero .

H: " (...) *the diet has to be good... Set it here...*" (they put the level of *diet* somewhere in the middle)

F: " *Did you start* (the simulation)*with it as good ? Always ?*" (Confirming whether their idea was to have *diet* starting from a value that means "good")

H: "*Yeah...*"

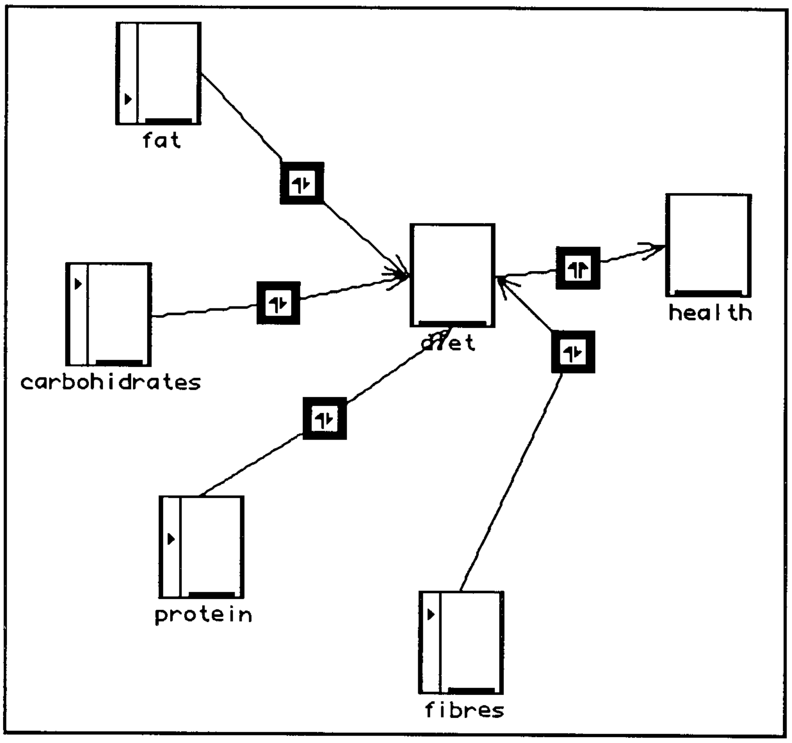


Figure 9.13: Group H: Model about diet and healthy Life. For the students the variable *diet* should begin with an initial value to represent the idea of someone beginning with a "good diet"

This passage happened with Group A during Task 7. They were trying to find a problem that fitted the structure below.

XXX --(**GTo,same**) -->

-- **add** --> ZZZ

YYY --(**GTo,same**) -->

They decided to work with a problem about a zoo where the different species of animals could make a better (or bigger) *zoo*. They thought about classes of animals such as *mammals, amphibians and birds* also suggesting the idea of a cock and a chicken mating and making a family. They connected both variables *mammals* and *amphibians and birds* to *zoo* using 'go together same direction' links.

When they went on to a simulation, they gave initial values to all variables. After running the model they were surprised with the fact of *zoo* ending at the same height whatever its initial condition. They wanted to represent the idea of *zoo* beginning with a certain number of animals and the causal variables (*mammals* and *amphibians and birds*) would contribute more to increase its value (add new animals to what already exists in the zoo).

F: " (...) *when we set zoo down there, then you ran it, it* (zoo) *went up there, didn't it ? In the other case you already had zoo at a certain height then you ran it and it also went up*"

C: " *What I was saying... the others are mammals, amphibious and birds. What it* (the system) *does, for instance, when it* (zoo) *begins in the middle, it would already have other animals, wouldnt it ? But it doesn't consider this. It assumes it doesn't have anything. It puts mammals, amphibians and birds.*"

Although this idea of "adding up" a certain value can be modelled by using the existing resources of the system (in the case above the students had to introduce a new variable that represented the "old zoo" and add it to the other animals - *mammals, amphibious and birds* - in order to calculate the new value of *zoo*), it is not straightforward. So, the addition of a "new feature" that permits the incrementation/decrementation of the actual value of a dependent variable when using 'go together' links, would make the system closer to the users' ideas, making it easier to model certain problems. However, it is not obvious that the idea is logically consistent with the rest of the system.

9.2.2.3 Strength of the Links

There were different reasons for changing the strength of the links. However the arguments used by the student could take into consideration only one cause and the dependent factor (even when more than one cause existed) or the relative relation between the different causes of a certain dependent factor.

In the first case the following general ideas were used as arguments to change the strength of a link:

• To implement a limitation (either to the causal factor or to the dependent variable);

• To delay the variation of the level of the dependent variable.

• <u>To implement a limitation</u>

The idea of changing the strength of a link in this case is to implement a certain limitation that has to be imposed either on the causal factor or on the dependent variable. This limitation can be related to the exceptions to a certain rule (first passage below) or to the implementation of a certain rule (second passage below).

> This passage happened with Group A during Task 6 (Constructing a model about "Diet and healthy life"):
>
> They were discussing about good and bad factors that causes *health* and decided to introduce a variable called *obesity* (it was caused by *fat and sweets*) also affecting *health* ('go together-opposite direction' link) and Lucia made a comment about the link between *obesity* and *health*:
>
> L: " *But there are many people that are fat and have a good health* "
>
> C: " *Well, we can make the link weak...*"
>
> L: " *OK* "

Because not all fat people are unhealthy was a justification to make the link 'weak'.

The next passage happened with Group E during the same task. Here the effect of the link was related to a limitation that had to be imposed on the causal factor in order to implement a certain rule ("people only have a good health if they have a good diet "):

> After constructing the model shown in Figure 10.15, I challenged Diego about the behaviour of diet when the person was eating a lot of fat. He then decided to change the link between them:

F: " *If you eat a lot of fat...the text says you can only eat 10 %...*"

D (correcting me) : " *5 %* "

F: " *You are eating a lot of fat and you* (referring to his model) *continues saying that the diet is correct...*"

D: "*Could change the effect* (of the link) *to weak. The correct thing is not to eat a lot of fat*"

• <u>To delay the variation of the level of the dependent variable</u>

In this passage Group A was discussing some ideas about diminishing the social problems in big cities (Task 4 - Creating a model about Migration to big cities) represented by the variable *problems*. The suggestion they made was to have some kind of social service which would construct houses and shelters for children and homeless. The general name they gave to this idea was *improvements.* They first changed *problems* to 'On/Off' (because, according to them, *improvements* only begins to happen after *problems* arrive at a certain level) and created a negative feedback loop between *problems* and *improvements* (both links were 'go together'). Before they went on to test it they decided to change both links to weak because

C: " (...) *improvements never include everything* (*Improvements* will not solve all social problems)"

(...)

C: "*When problems start, improvements do not happen very quick. Otherwise there wouldn't be people living on the streets today*"

Because *improvements* does not start soon after the *problems* arrives at a certain level, a 'weak' link has to be used to "delay" the improvements.

In the case where the relative relationship between different causes was taken into consideration, the justification had the following patterns:

• One variable is not as important as the other;

• To change the effect of one causal variable on its dependent variable. More or less like making the constant K in the equation $y = K * x$, different from 1;

• The effect of one variable on the causal factor will never cancel the effect of the other one.

• <u>One variable is not as important as the other</u>

This was the most common reason for changing the strength of a link. However, this idea of "more importance" could simply be a justification for making the model behave in a certain way (the passage below is an example where this happened):

Group D was constructing a model about "Diet and healthy life" (Task 6). They had already constructed the model shown in Figure 10.1 and were about to try a first simulation with it:

L: " *Set good food in the middle and bad factors in the middle as well...*"
M: (correcting what Luis said) "(bad factors) *stayed a bit above* (the middle)... *lets see what happens...*"

They ran the model and saw *health* equal zero...

L: " *Yes. that is it, setting bad factors above good food you are not going to have health, are you ? ...* (you)*Have to diminish this* (bad factors)..."
M: (after running the model again) " *Just a bit of health and a bit of life expectancy*"
L: " *Lowering bad factors, putting it below the middle...*"
M: " *You're going to have not much health... not much life expectancy...*"
L: " *That is why I think that* (we have to make) *bad factors with a weak relation...*"

They changed the link, ran the model again with the same initial conditions and saw *health* and *life expectancy* not very low. This time they accepted the model.

The next passage is also an example where more importance is given to a certain variable. In this case the justification for this importance is implied by a certain action that should be taken by the person represented by the model.

This passage happened with Group A during Task 6 (constructing a model about "Diet and healthy life").
They were discussing good and bad factors that cause *health.* After creating the variables, they began to connect them. Because it was stated in the text that *grains and vegetables* should form a high percentage of participation in someone's diet in order to achieve good health, they used a strong link to reflect this. The other links did not need to be modified because *fat and sweets* was alone in the bad group and *fruits meat and dairy products* was already different from *grains and vegetables* :

C: " *Grains and vegetables is stronger in order to differentiate...*"

F: " *With stronger are you meaning it is more important ? To* (determine) *health ?*"

C: "*Yes. Because it is the most important* (...) *Yes. All are important* (but) *what you have to eat more is grain and vegetables*"

In the next two passages it seemed to happen rather differently. In these extracts, the level of importance of a certain variable in the problem was the only necessary condition in making changes in the strength of the link:

Group D was working with the model about "income of the waiter" (Task 3 activity 1). We had created the model with three variables: *salary*, *tips* and *total income* (Both *salary* and *tips* were connected to *total income* via a 'go together-same direction' link) and tried some simulations with it. They then made a suggestion:

M: " *Isn't it better to set tips causing a weakened effect ? Because you don't have a tip so high that you can compare* (to put on a level) *with salary* "

The idea here seemed to be "whatever you earn, salary always has more importance in determining your income".

Later in this same task they suggested a similar problem about calculating the retail price of a product. It had two causes: *wholesale price* and *tax*. Both were connected to *retail price* via 'go together-same direction' links. When they connected the variables they commented:

M: " *When wholesale price doesn't increase the retail price doesn't increase, the tax increase, the retail price...it is the same thing* (similar to the previous problem about the waiter), *the tax could have a weaker link* "

L: " *It doesn't count for a lot*"

The idea here seems to be however much *tax* you are paying it will never be more important than *wholesale price* to determine the *final price* of a product.

• <u>To change the effect of one causal variable on its dependent variable.</u>

In this passage Group H was working with the model about "Accidents on the road" (Task 3 activity 2) and Fabricio was about to introduce the variable *use of safety belts* . He decided to connect it to *accidents with death* via a 'go together-opposite direction' link to represent the idea of "safety belts reduces the number of deaths in road accidents". He first used a 'normal' weight to connect it to *accidents with death* but after running the model, he noticed that *accidents with death* was very low even though *total number of accidents* (another cause of *accidents with death*) was very high. He then decided to change the weight of the new link to 'weak'.

F: (after running the model) " *The total number of accidents is up there but accidents with death continues zero...*"
Fa: " *I think it is... I think it has to set it weaker...*"
F: " *The link between use of safety belts and accidents with death ?* "
Fa: " *Yeah...*"
F: " (...) *what do you mean by it ?*"
Fa: " *It reduces, but it doesn't* (have to) *reduce very much.*"

This passage could also be seen as a case of using the strength of the link to implement a constraint imposed by the real world (Here the constraint could be stated as: "in some cases the use of safety belts reduces the number of deaths").

• <u>The effect of one variable on the causal factor will never cancel the effect of the other one</u>

This passage happened with Group C in the beginning of Task 3 activity 3. I constructed the model about the bath tub to introduce the idea of 'cumulative' links. I did some simulations using different values for the *tap* and *drain*. In the last one, the bath tub was almost full and I set *tap* and *drain* to maximum and they noticed that the level of the tub did not change. I then started asking questions about the problem:

F: "*What could be done to simulate the idea that the tap is much bigger than the drain ?* "
R: " (you) *have to set here, it would be 'strong' here in the tap.* "

At least two answers could arrive from my question. They could say something either about changing the level of the causal variables or about the strength of the links. I think they chose the strength of the link because both levels of *tap* and *drain* were already maximum in the last simulation. Therefore to make *tap* bigger than it was could only be possible by changing its effect on the *bath tub.*

Although this passage can be seen as an example of representing the idea of something being more important than another, it also carries with it the idea of things counteracting each other (or not). Nevertheless, in the next passage the idea of factors being compensating is more clear cut.

> This passage happened with Group H during Task 8 (Constructing a model about "Pollution in big cities"). Fabricio began with the idea of different types of pollution being represented by the same box. The factors that contribute to increasing pollution were put on one side of the screen and the ones that contribute to alleviating it were put on the other side (see Figure 9.14).
>
> When Fabricio began to create the links, he decided to 'weaken' the ones departing from *street cleaners* and *Sewage Department* to *pollution* because:
>
> Fa: " *the sewage system and the street cleaners are not enough to clean up so much pollution*".

Here the strength of a link was not used to indicate that a certain factor is less important than the others but to mean that it will never be enough to compensate the others.

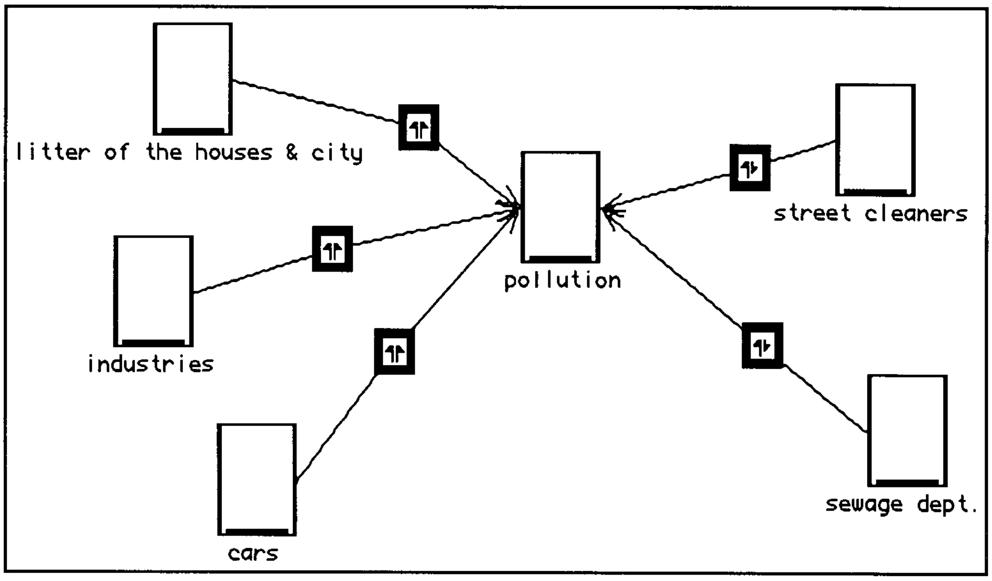


Figure 9.14: Group H: First model about pollution

9.2.2.4 Using 'same' and 'opposite' Directions

'Same direction' relationships seem to be the most natural way of thinking about cause and effect (about a quarter of the links created during expressive tasks were 'opposite direction' links). One reason may be that students brought a lot of their knowledge about the real world to the tasks they performed with LinkIt and there, in the real world, they are much more accustomed to talk about "same direction relations". It is also true that one can (in most cases) convert opposite direction relationship to same direction by simply reversing the meaning of the causal factor.

Although not very common, there were situations where the students associated the idea of 'same direction' links with dependent factors increasing and 'opposite direction' with dependent factors decreasing:

> This passage happened with Group A during Task 5 activity 3 when constructing a model about a heating system.
> Their model had 03 variables: *outside temperature*, *house* (representing the temperature inside the house) and *heating* . They wanted temperature of the *house* to follow *external temperature*. But because at the same time they were thinking about *external temperature* falling (to represent the idea of getting cold) they suggested an 'opposite' link between them.
>
> C: " *It has to be opposite here doesn't it ?* "
> F: " *Opposite where ?* (between) *outside temperature and house ?*"
> C: *Yeah, it is true, the smaller the* (outside) *temperature, the smaller the temperature of the house* "

Nevertheless, what happened more often was to analyse the relationships only taking into account a partial behaviour of the causal factor. It seems that in most of these cases they did it intentionally. One example happened during the construction of a model about a heating system (Task 5 activity 3) where Group D created a variable *cold* connected to *inside temperature* via an 'opposite' link. They used the name "cold" because they were analysing a situation about heating systems which only "makes sense" in a cold environment. They used 'opposite' link because *cold* temperatures make the *temperature inside the house* drop:

> L: " *Do you think that outside temperature of the house could be cold ?*"
> M: " *Because it could keep diminishing the inside temperature of the house... we could put an opposite ball* (cumulative) link..."
> F: "*From outside* (temperature) *to inside, is it ?*"
> L: " *Yes, instead of using outside temperature,* (we can) *use the box as cold, because the cold would diminish the inside temperature of the house...*"

Using 'opposite' as 'inverse'

LinkIt permits two kinds of 'inverse'. One is mathematically represented by $Y \approx 1/X$ and can be achieved by combining 'multiplication' with 'opposite' links, and the other is $Y \approx MAX - X$ (MAX is the maximum value of a variable within the system) which can be achieved by having two or more variables connected to a dependent variable (Y) via 'go together' links with at least one 'same direction' and one 'opposite direction' (which is the link departing from the variable X). The range of the dependent variable does not affect the idea of working as 'inverse' although if it is 'any value' its visual behaviour on the screen will be more in accordance with the idea of 'inverse'.

This second possibility of using 'inverse' was presented to the students at the very beginning of the experiment, during the construction of the model about the "Income of the waiter" (Task 3 activity 1). There the variables *salary* and *deductions* were connected to *total income* via 2 'go together' links ('same' and 'opposite' direction respectively). At that time, the range of *total income* was also changed to 'any value' to represent the idea of "borrowing money from the bank" which could possibly be used by the students to reinforce and generalize the idea of "go together-opposite direction links work as inverse". So, when they tried to use it in other situations where they thought the "rule" could be applied, they were surprised at the simulation and tried to "fix" the problem by using different artifices such as changing the type/name of a variable or type/direction of a link involved in the problem[4]. The example shown below is a case where they changed the type of a variable:

> This passage happened with Group H when constructing the model about Migration to big cities (Task 4). At this time their model had three 'smooth' variables. *Rural population* and *population of the city* were connected to *jobs on offer* via two 'go together-opposite direction' links. They justified the links saying:
>
> H: " (...) *there is a certain number of jobs in the city and when more people are entering* (moving to the city), *less jobs* (will be) *on offer... so I think that jobs on offer had to diminish according to rural population and population in the city increase...*"
>
> After trying some simulations they said:
>
> H: " *It* (jobs on offer) *always goes down...it keeps always going down...it doesn't make sense... It has to have a condition... if population of the city*

[4] It is important to mention that this situation was not found during the work with IQON . It seems that there this problem was hidden by the fact that all variables were 'any value'. So when the cause was big, the dependent factor would go below zero.

plus rural population ... until a certain level jobs on offer is OK... when they cross that point, jobs on offer will fall..."
Fa: " *So put* (use) *the On/Off box*"

The next passage is an example where the idea about using 'opposite' as 'inverse' is more clear.

This passage happened with Group J during Task 8 when constructing a model about pollution.
When they began to construct the model, they changed slightly the focus of the problem from a city to a country (Brazil) and according to them, there were two central points in this problem: the consciousness of the first and third worlds (see Figure 9.15).
When they went to test their model I realised that they were seeing the relation between *consciesness of the first world* and *pollution* as an 'inverse' relation:

F: *Are you seeing...(...) when consciousness of the first world is down there what do you want to happen to pollution ?*'
V: " *High*"
F: " (...) *and when consciousness of the first world is up there where do you want pollution ?* "
V: " *Low...*"

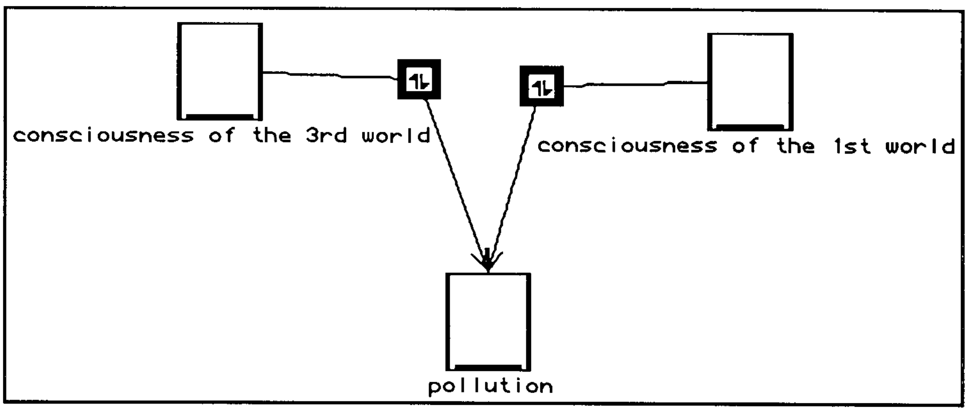


Figure 9.15: Group J: Model about pollution. Seeing the relations as 'inverse'

It is also true that in our daily experience a clear distinction between the ideas of "opposite" and "inverse" is not made. However, using the system, at least in one special situation, the user has to make this distinction: when there is only one 'go together-opposite direction' link arriving at the dependent variable.

The discussion above also serves as a partial justification for the fact that the students scarcely explored the use of 'multiplication' with 'opposite' links (see Topic "Use of multiply" in Section 9.2.1.8 and Section 9.2.3.3). They simply thought it could always be achieved simply by using 'opposite' links.

9.2.2.5 Use of State of a Link

Although students were curious about the functionality of this property (In the beginning of the work with LinkIt, when they were inspecting the Information boxes, they used to make questions about it), it was not used very much. The main reason seems to be the fact that they assumed that making a link 'asleep' would automatically make a variable 'asleep' as well. Therefore they tended to change it in the variable.

9.2.3 Other Aspects of the Interface

As expected from the Preliminary study and the modifications made as a result, the students did not have problems in interacting with the interface. From the beginning they understood the explanations about how to manipulate the objects and in some cases they talked about the similarities between this system and other computer programs they already used.

Some problems that were reported during the pilot study (see Chapter 5) were not noticed here. Particularly noticeable was the fact that during that study some students preferred to create a new model instead of deleting/changing the name of some variables because the system used at that time (Prototype I) would take too long to redraw the model on the screen. Some reasons for this kind of attitude not previously reported in this study are the improvements made in the algorithms to calculate whether a link/variable should be redrawn and the use of a more powerful (and therefore faster) computer (IBM PC 486 compatible - 66/33 MHz).

Below some particular aspects of the interface that either did not exist in Prototype I or are worth being commented on are presented.

9.2.3.1 "Dragging level while it is running"

This feature was implemented in Prototype II mainly to improve students' control over the simulations and indeed it was used by them during the tasks to demonstrate/explain situations while the model was running. An example happened with Group C during Task 5 activity 2 (Model about the "Eye-pupil") when they had to answer question 2 (about representing the idea of "before and after entering the tunnel"). They started a simulation with a high level of *amount of light* (to represent the idea "before entering the tunnel") and while the model was running they began to drag the level of *amount of light* first down (to represent the idea of "being inside the tunnel") and after up (to represent the idea of "getting out of the tunnel"). During this process they kept observing what was happening to the variables (concentrating on *size of the pupil*) and explaining what would happen to the eyes in a real situation.

9.2.3.2 Use of the Control Panel

Although the students had no problems in understanding and manipulating the buttons related to the running operations, I observed that during some simulations they wanted to start from the same initial conditions but making small changes to the levels of some variables (normally one or two variables). Because the system does not give any means to do this, the students had to spend some time adjusting the model before a new simulation. Therefore it would be helpful if it was possible to make the model go back to the initial conditions set by the user, helping them to try new simulations. This could possibly be done by creating a new button to perform this operation. In this case it would also be interesting to maintain a "circular list" of a certain size "n", with information about the last "n" simulations. The user would simply have to click on the new button to have the variables set to the initial state of the last simulation, or a previous one.

9.2.3.3 Combining Links

Adding values and calculating the inverse were by far the most natural ways for the students to represent their ideas with LinkIt. Because they tended to see these two operations as complementary, they tended to implement the inverse by using 'go-together opposite direction' links with addition. The problem is that this type of relation permits calculating the inverse only in special circumstances (see Topic "Using opposite as inverse" in Section 9.2.2.4 for a discussion about calculating the 'inverse') and the students in most of their work did not "pay attention" to its constraints. The other possible way of calculating the inverse (using 'multiplication' with 'opposite' links) is located in another part of the system - for reasons of following a mathematical coherence between the operations - which they did not explore very far. Therefore they do not appear to be doing anything wrong, indeed they are simply applying a very basic rule of interface design where "similar ideas must be close to each other" (Shneiderman, 1992).

The students could be helped to employ the appropriate inverse by designing particular examples (to be presented during a learning phase) where the differences (and constraints) between the two ways of calculating it are stressed.

Also the fact that the students had to change the 'input combination' of a variable through one of the Information boxes of the incoming links (and not in the Information box of the variable itself), made some of them confused about which links were being combined. In some cases they also asked whether it was necessary to change this property in other links as well. This problem could possibly be reduced just by changing the place where the 'input combination' is modified from the Information box of the links to the Information box of the variables.

9.3 CONCLUSIONS

The students were able to understand and use the system's interface to construct and explore models about situations of the real world. Although sometimes they tried to employ LinkIt's objects in ways not predicted by the system conceptual model, they (in some situations) went beyond its implementation, suggesting new features that - if incorporated in the system - could widen the range of problems to be modelled with LinkIt.

Particularly the passages above provide evidence that the students, after a brief demonstration of how the main objects of the system work, are able to use them to construct and explore models.

It is also relevant to point out that the students did not learn everything about the system at the first meeting. During the second, third and fourth meetings they took many steps forward - and some back - in their understanding of the system. This behaviour is quite understandable, especially if we bear in mind that the students were engaged, at the same time, in two demanding tasks: learning how to operate a new tool, and using this tool to construct and explore models.

Chapter 10:

ANALYSIS OF STUDENTS' THINKING AND LEARNING WITH LINKIT

10.1 INTRODUCTION

This chapter presents a discussion of how students explore and create LinkIt models.

As we shall see below, the students had the opportunity to go further in their ideas when interacting with LinkIt to think about and explore certain domains related to their school science curriculum.

It is also interesting to note how much of their knowledge about the real world they employed to justify, explain and reason about the modelling activities. To some extent this behaviour was predictable, mainly because this was - most probably - the first time they had a chance to engage in such modelling activities within their school activities.

10.2 ABOUT STUDENTS CREATING AND EXPLORING MODELS

This section contains six subsections, each discussing a relevant aspect about the behaviour of the students when creating/exploring models. They are:
- The purpose of a model;
- Evaluating a model: What they did and their reactions;
- Patterns in the construction of models;
- Recognising analogous model structures;
- Relating models and the real world;
- Making predictions and testing them.

10.2.1 The Purpose of a Model

Although at any moment the students were specifically asked about the purpose of their models, the interpretation made is mainly based on their observed behaviour when approaching the tasks.

The purpose of their models was very much influenced by what was demanded by the tasks. Most of the expressive tasks asked them to express their ideas about a certain problem. In these situations their behaviour (and their models) were much like to explain/demonstrate an idea. Some questions in the worksheet of the exploratory tasks challenged them to make hypotheses and test them. In these situations they most often approached the models to test their ideas about the problems. However, at least two particularly interesting variations from this behaviour was noticed [1]:

- To show different situations of a problem;
- To calculate a final result (model used as a calculator).

• <u>To explain/demonstrate an idea</u>

This way of using a model is characterised by the students trying to explain a certain situation using the model to support/visually demonstrate their ideas. An example of this behaviour happened with Group C during Task 5 activity 2 (Model about the "Eye-pupil"): The students had to implement the idea of someone going in and out of a tunnel (answering question 2 on the work sheet). They started a simulation with a high level of *amount of light* (to represent the idea "before entering the tunnel") and while the model was running they began to drag the level of *amount of light* first down (to represent

[1] It is also worth mentioning that some students, during the same task, manipulated a certain model with different purposes. A typical case is when they were constructing a model to represent a certain idea and they faced a variable behaving in an unexpected way. At this time they shifted from a broader goal (e.g. to represent the idea about causes and consequences of people moving to big cities) to a "localized" one where they had to do some tests with variables and links in order to "fix" the problem.(See topic ahead: "Patterns in the Construction of Models").

the idea of "being inside the tunnel") and after up (to represent the idea of "going out of the tunnel"). While the model was running they kept explaining what would happen to the eyes in a real situation.

However there were cases where the students explained an idea using the model simply as a "static drawing", not exploiting its dynamic possibilities. An example happened with Group J during the same activity, answering the same
question mentioned above : Fabricio devised an experiment which he explained while setting all important variables accordingly. Only after "receiving" my agreement, did he decide to run it.

• <u>To show different situations of a problem</u>

A model could be used to show different aspects of a problem. An example is shown in the next two Figures below. The model about "Diet and good Health" created by Group D (Task 6) was used to "calculate" how good or bad someone's health is (Figure 10.1) or to demonstrate how someone's health changes with time (Figure 10.2). A change from one interpretation to the other is just a matter of changing the type of some links (normally from 'go together' to 'cumulative') and variables, also including (sometimes) the creation of feedback loops.

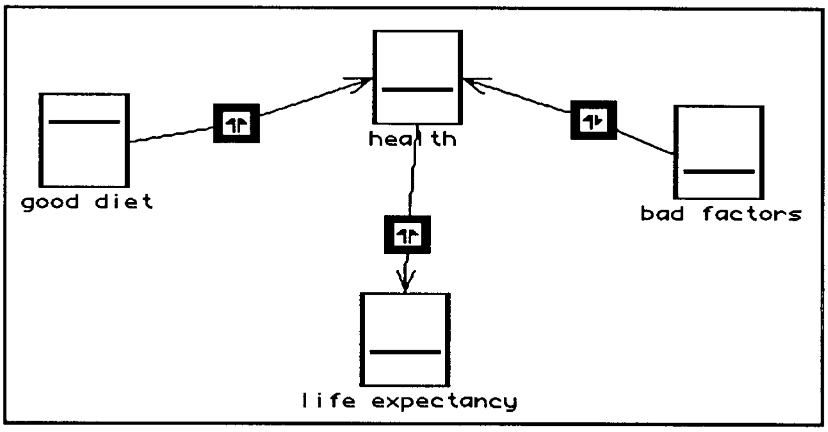


Figure 10.1: First model about health and good diet (Group D): The model was mainly to 'calculate' how good or bad is someone's health

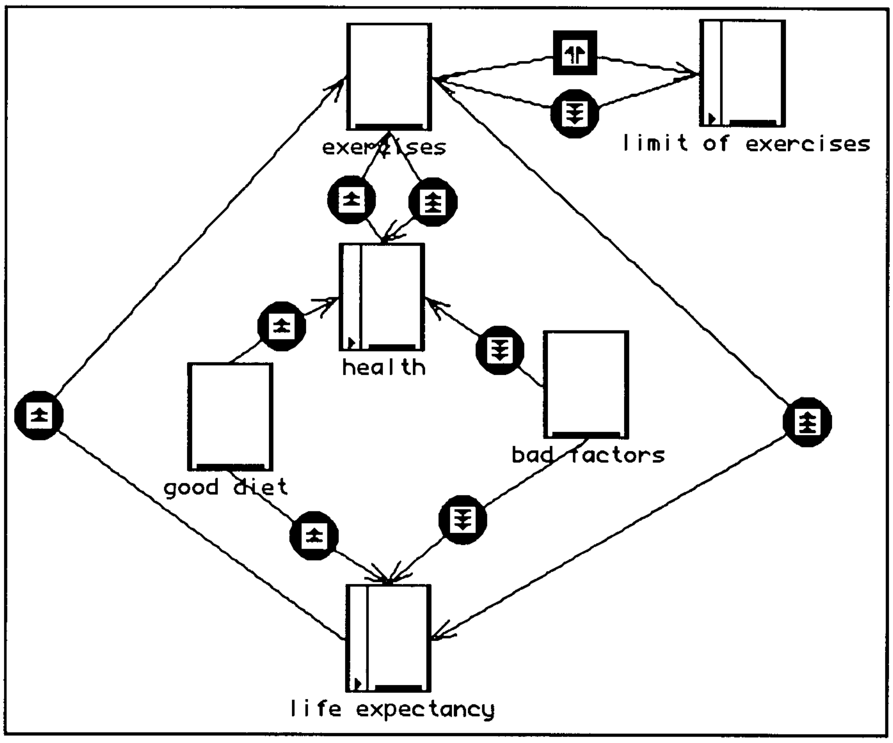


Figure 10.2: Last model about health and good diet (Group D): The model was mainly to show someone's health changing over time

• <u>To calculate a final result (model used as a calculator)</u>

The main purpose of using the models in this way is to employ them to compute a certain value after some initial values are given. Normally these models are mainly composed of 'go together' links and have a "star-like" format[2].

An interesting example happened with Group J during Task 6 where they used their model to test their own health (see Figure 10.3):

> After trying a simulation for her own diet - and seeing the model not giving a "good result" - Vera started saying that she knew she did not have a healthy diet and that she had already started to eat more vegetables and fruit and to take less cola drinks[3].
>
> V: " (...) *I am trying to get used to eating more vegetables...I used to eat only meat, rice and chips... even beans I didn't want to eat... I used to eat at least one plate of chips per day... I used to eat an average of 3 eggs... drink 1 litre of coke... Nowadays I don't even drink coke anymore* (...) *Because I used to drink a lot* (of coke), *I can't drink it any more*"

[2] A model has a star-like format when there are many causal factors arriving at the same dependent factor. The model shown in Figure 10.3 is a typical example.

[3] This conversation also made them bring to our discussion a topic they had in a biology class on the day before about heart disease and fat (See topic ahead: "Relating Models and the Real World").

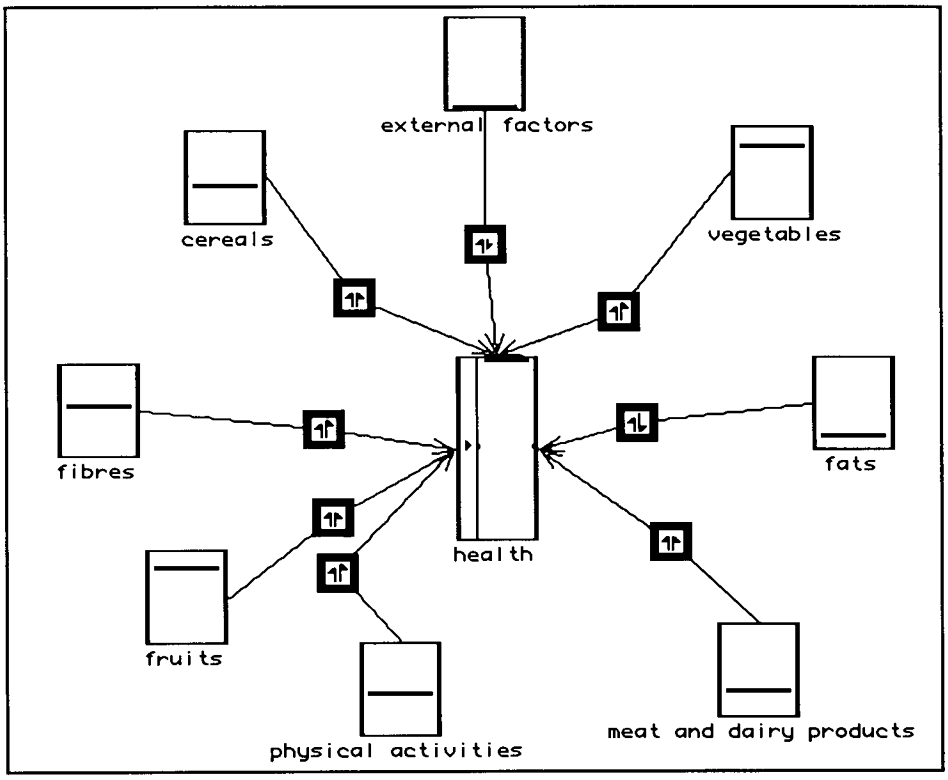


Figure 10.3: Model about diet and health used by the students to 'investigate' their own health (Group J)

• <u>To test some ideas</u>

This case is characterised by the students devising some hypotheses/predictions and using the models to test them[4]. These situations were mainly observed during the exploratory tasks where the questions in the work[illegible]sheet led them to behave in this way (For more about students making predictions and devising tests, see section 10.2.6).

An example of making predictions and using the model to test them happened with Group D during the work with the model about the refrigerator (Task 5 activity 3):

> During our conversation they told me that the refrigerator keeps functioning longer if the *outside temperature* is high. I asked them how they could prove their ideas were correct and they devised an experiment in which we could observe the system clock during two different events. One with the *outside temperature* very high and the other one with it low. According to their prediction in the first case the *thermostat* would take less time to be triggered.

---

4 Although the transcriptions are full of examples where the students found something not going very well (e.g. a variable not behaving in the way they wanted) and made some tests to try to "fix" the problem, I do not consider these cases as examples of using the model to test some ideas because they are "localized events" that the students had to overcome in order to achieve a broader goal (See topic ahead: "Patterns in the Construction of Models").

F: "*... and if I take this refrigerator and move it into the sun* (outside the house). *What do you think would happen to the model ?*"

M: " *The temperature inside the refrigerator would increase faster*"

(...)

F: " *What can you do to convince me about your ideas ?*"

M: " *You can reset it here and see how long it takes with each...with sun and without sun*" (the sun is connected to outside temperature via a 'go together-same direction' link)

Another example happened with Group E during Task 5 activity 2 (Exploring a model about the "Eye-pupil"). The model I showed them had the direction of two links changed from that in the original task (between *excess of light* and *size of pupil* and between *size of pupil* and *light in the eye*) in order to make the variable *size of pupil* behave in the wrong way (see Figure 10.4). In this example, because the model was not in accordance to their ideas, they had to change it to make it behave in the way they expected.

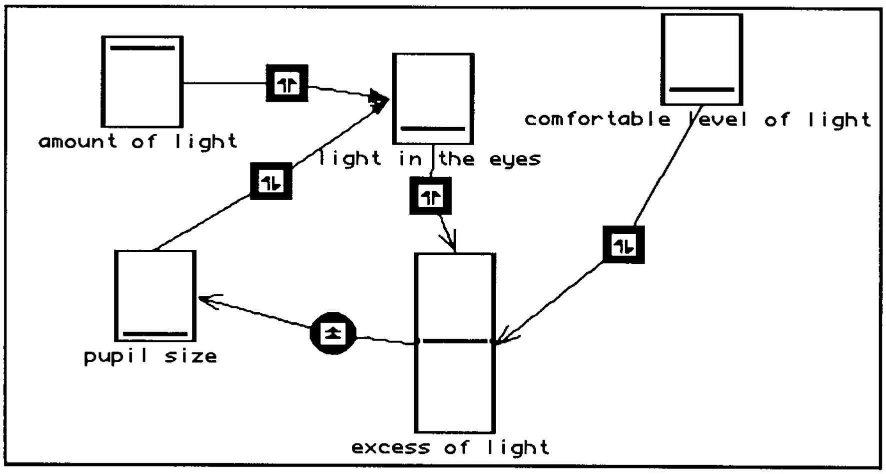


Figure 10.4: Model about eye-pupil presented to Group E

Before running the model they were already expecting to see *pupil size* go down. To test their idea they simply set the values of *amount of light* (very high) and *comfortable level of light* . However, during the simulation, the model did not behave in the way they wanted:

D: " *It is wrong.* (the behaviour of size of pupil)*..It should be the contrary, it should be low... If you have a lot of sun your pupil is going to be smaller... if you're in the darkness, it increases. When you're in a bright place it diminishes.*"

Because the model was not according to their expectation, they attempted to "fix" it by making changes in the link between *size of pupil* and *light in the eyes* :

D: "*This link here is wrong... from pupil size to light in the eyes...*"
P: " *Make it equal* ('same direction') " (This change was one of the two necessary changes that had to be made to make the model work correctly)

10.2.2 Evaluating Models: What They Did and Their Reactions

During tasks 4, 5, 6 and 8 the students had either to explore or construct a model. All of these tasks (except Task 5 activity 3 and Task 8) had an introductory text that presented some ideas about the topics they were going to work with. During the work with tasks 5 and 8 students also had a worksheet where some questions were presented. Particularly in the 3 activities of Task 5 (exploratory activities) the worksheets were prepared to "stimulate" the students to explore the models presented to them. Also at the end of each of these tasks I asked them whether they thought the model they worked with was, for them, a good or bad representation of the problem in the real world.

What the students did and their reactions about the models seemed closely related to the kind of tasks they were engaged in and their knowledge about the domain involved in these tasks. For that reason, I have decided to discuss these subjects in two different subsections. One related to the exploratory tasks (Task 5 activities 1, 2 and 3) and another to the expressive tasks (Task 4, part of activity 3 in Task 5, Task 6 and Task 8).

Exploratory tasks

Students had a tendency to criticise the models used during the exploratory tasks in terms of what was said about them in the introductory texts. This behaviour was specially common during the work with the model about "predator-prey" (Task 5 activity 1) where it was very difficult for them to find a set of initial values where the model could behave in the way suggested by the text.

Another important finding was that the students did not go further in their tests in order to discover whether the model could be used to demonstrate what was suggested in the text. Normally one simulation at the beginning of the activity was enough to them to come up with a conclusion. In these cases I had to draw their attention to the fact that they were only making partial tests and that therefore their conclusions could be wrong. Below are presented some examples that illustrate these ideas:

After the first simulation with the model about "predator-prey", Group D said:

F: " *What did you see ?*"

M: " *Rabbits increasing and foxes also increasing...*"

L: " *But before this the rabbits...*"

M: (completing what Lucio was saying) " *Increased and foxes dropped*"

(...)

M: " *This model is not showing what we read: rabbits going down and foxes increasing...I think it is exactly the contrary* (of what the model is showing) *because foxes eat the rabbits, rabbits keep diminishing... population of foxes increase...and because the population of foxes is much bigger than the rabbits, they fight against each other, the foxes, and they diminish... many die and the rabbits can increase*"

Later, after I told them that they had only tried one initial condition for the simulation, they made other tests with the model and when I asked them about the model they said:

L: " *We made the three tests* (the three possible combinations for *population of foxes* and *rabbits*) *and in all of them there was much more rabbits than foxes...*"

M: "*It is not matching with what the text says.*"

Other groups like C and E also used the text to "approve" the model:

At the end of the same activity, Group E said about the model[5]:

D: " *Because it represents exactly what is said in the text...When there are lots of rabbits.... no, when there are rabbits and foxes* (...) "

P: " *It is representing the same as the text* "

D: " *Yes. Because it is representing exactly what is said in the text..*"

Another example happened with Group E during activity 2 (Exploring a model about the "Eye-pupil"). After only one simulation they said [6]:

D: " *It is wrong.* (the behaviour of size of pupil)*..It should be the contrary, it should be low... If you have a lot of sun your pupil is going to be*

[5] Athough the model was shown to this group with minor changes in order to make it behave in the wrong way, they managed to make it behave in the way suggested by the text.

[6] See the second passage in "To test some ideas" under the Section 10.2.1 - "The purpose of a model" for a detailed description of this Group working in this activity.

smaller... if you're in the darkness, it increases. When you're in a bright place it diminishes."

Another approach to judging a model was discussing how much of its structure reflects the ideas of the text. Group A, when working with the same task, behaved in this way:

They did not agree about whether the model was representing reality. For Lucia it was OK because "*it is representing the natural equilibrium*" but Carmen did not accept her argument replying "*No. All species died*" . It seems that each was focusing on different aspects of the model. Lucia was paying attention to the structure of the model -saying that its structure represents the natural equilibrium between the species in the nature - which was in accordance with the text (the main variables and relationships suggested by the text were represented in the model). Carmen, in her turn, was more concerned with the behaviour of the model (its dynamics) and in that case the model did not correspond to what they read. A bit later (after Lucia mentioned that the question was about the model representing reality) Carmen agreed with her opinion but said it was not in equilibrium (talking about the model running). After that they did another simulation and this time they saw the model in equilibrium. Then they wrote down in their worksheet that the model was correct because " *it is representing the situation about reality* (...)"

It seems that when the students gain confidence or know more about a certain domain their attitude towards the models changes. They normally become more willing to make changes in the models in order to improve or "fix" them. Also their ideas about the models become more sophisticated, as they include other aspects brought from their daily-life experience. An example that supports this idea is from Group D, working with the three different activities in Task 5:

After understanding and accepting the model as a representation of a refrigerator working (Task 5 activity 3), they decided to make some changes in the model in order to find a way of eliminating the variable *gain from outside* from the model. They did it by putting the variable *gain from outside* and all of its outgoing links to 'sleep' and running the model for some different initial conditions. They noticed some minor differences in the behaviour of *thermostat* but, for them, the system was still working well:

L: " *I think it is working correctly, though the thermostat is going beyond* (the trigger level) a bit (before it turns on)..."
M: " *Yeah. But soon after it decreases...*"
L: " *But it isn't even very much... it goes a bit beyond its level but soon after it decreases...*"
M: " *It wouldn't lose the food...*" (this new behaviour of the thermostat would not make the food to defrost)

Two models earlier when they were working with the problem about "predator-prey" (see the first passage under this topic) they did not accept that model as a good representation of reality because it was not functioning exactly as they read in the text. Now in this task the model, after some changes, was not functioning in the same way as in its first version (which they accepted as correct) and they still thought it was right. Why ?. A possible answer can be given if we compare the analysis of activities 1 and 2 and what the students said at the end of activity 2:

During activity 1 they were not willing to make tests with the model and relied very much on what was said in the text to answer the questions in the worksheet. However, their behaviour during activity 2 was quite different. In this case they made more tests, even affirming (correctly) that the model was not working well. By the end of this activity I was very puzzled about their different attitudes when working with these two models and decided to ask them why:

F: " *In that problem about predator-prey when you read the text you said the text was correct and the model was wrong. In this problem here you thought that...*"
M: " *The model was wrong as well...*"
F: " Yeah but the text doesn't say anything whether the model is correct or not (...)"
L: " *Yeah but in that case of predator-prey I didn't know very much about the subject, but here I have already heard about it. This thing about the eye-pupil that always...when there is more light the pupil closes more, I have already heard about this subject, but the other not.*"

In the case of the "predator-prey" problem they did not know very much about the problem and they were given a text which they accepted as telling everything about the problem. During activity 2 they knew more about the problem and therefore they were more confident about making changes and trying new things. In the same way, during activity 3, although there was nothing telling them how a refrigerator works, they demonstrated that they have some knowledge about the problem. So here again they relied on their own ideas about how a refrigerator works to make a judgement of the model based on its qualitatively correct behaviour saying things like " (even when the *thermostat* is taking a bit longer to turn on, it) *wouldn't lose the food* ".

Expressive tasks

The students approached expressive tasks quite differently from the exploratory tasks. Because they had to construct a model from the beginning, they only externalised their opinions about them when they reached what they considered a final model. During the process of constructing the models, their comments about them were more about the unexpected behaviour of certain variables or links or to give some justifications for some modifications they had to make.

The models the students constructed (entirely or partially) were never evaluated as "bad" or "wrong", they always served to represent part of a reality the students were talking about. An example is shown in the last passage above with group D where they used the argument "*it wouldn't lose the food* " to justify a different behaviour of the *thermostat* and to continue accepting the model as "good"[7]. Another example where they had to construct the model from the beginning, happened with Group A during Task 4 (Constructing a model about "Migration to big cities"). After failing to make the model behave in the way they wanted, they used a justification from the real world to continue accepting it:

> After they had constructed a model to represent the problem about migration, I asked them " *what could be done to diminish the problems of the city ?* ". The suggestion they came up with was to have some kind of social service which would construct houses and shelters for children and homeless. The general name they gave to this idea was *improvements.* They first changed *problems* to 'On/Off' (because, according to them, *improvements* only begins to happen after *problems* reaches a certain level) and created a negative feedback loop between *problems* and *improvements* (see Figure 10.5).
>
> After running the model and seeing that problems did not diminish, they associated this fact with their real life experience where this kind of improvements does not make too much difference:

---

[7] Although here is an exploratory task, at this moment students were engaged in an expressive activity trying to create a new model from the one shown on the screen.

L: " *The problems were not solved.*"

C: " *Because more people are coming*"

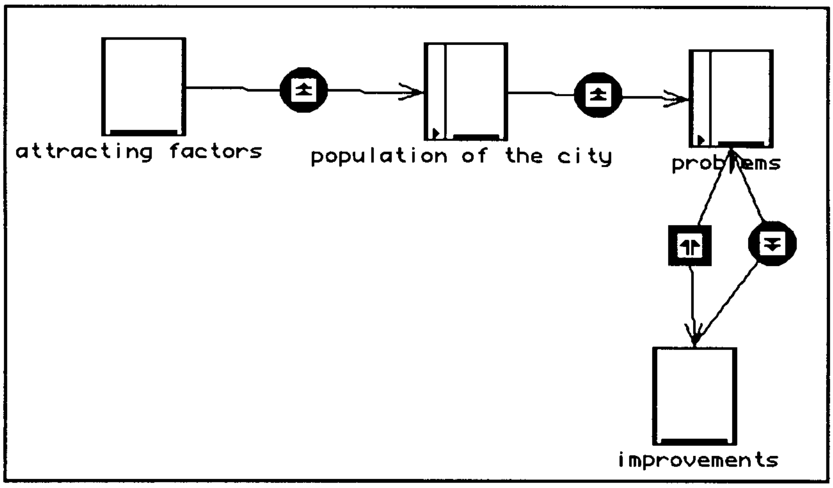


Figure 10.5: Model about migration to big cities (Group A)

Also Group I when they arrived at the end of Task 8 (Construct a model about "Pollution in big cities") without managing to make *pollution* stay at a reasonable level, considered their model as "utopia", meaning that it was showing something that could hardly be seen in a real life situation:

M (after a simulation to observe the variable pollution) : " (...) *because now you don't have pollution anymore, you don't have any kind of pollution... if it* (the real world) *was like that...but it isn't* (...)"

...

F: " *What did you think about this model ?*"

M: " *It is utopia...it is the ideal pollution*"

10.2.3 Patterns in the Construction of Models

In most cases when constructing a model, the students began creating a list of variables on the screen and afterwards went on to connect them. This behaviour is very much in accordance with what Richmond et al (Richmond, 1987) call "laundry list thinking". The idea is that when the students approach the modelling process, they first make a list of relevant variables necessary to describe the problem.

This way of constructing models can also be seen as a process of successive refinement where the students rely on the visual feedback of the system to improve their model successively, adapting them to their ideas. Roughly this process can be divided into three stages. First they simply want to produce a "draft" of their ideas, putting on the screen the relevant variables and their causal relations (creating the "laundry list"). After, they move to a second stage where they start doing tests and simulations in order to make variables and links work according to their ideas. At this time the focus of their attention is very localised, concentrating on the behaviour of variables and links. In that way, depending on the problem, it is possible to observe the students making a lot of changes in the types of variables and links and sometimes creating new variables either to implement a certain mechanism (such as a modelling gadget) to help fix part of the model or to introduce a sophistication into the model. During this interaction with the model they gain more confidence in what they are doing and new ideas begin to emerge. These are the starting point of a third stage where the students are able to demonstrate a further understanding of the problem and/or engage in increasing the sophistication of their models (see Chart 10.1).

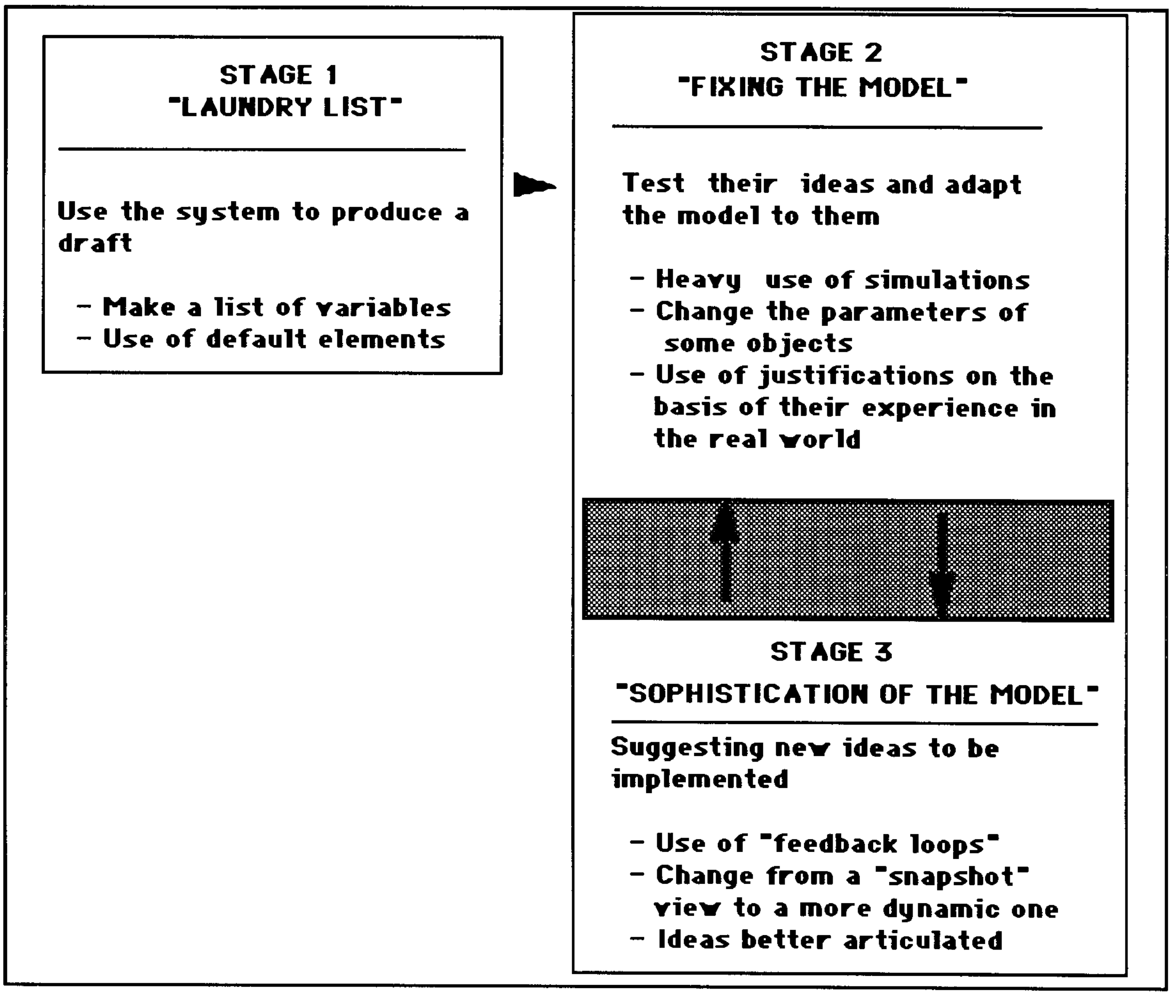


Chart 10.1: The three steps followed by the students when constructing models with LinkIt

An example of this idea is shown in the passage below, from Group J during Task 4 (Constructing a model about "Migration to big cities"):

They started the task by discussing about the main ideas of the problem and, at the same time, putting some variables on the screen:

V: "*Big cities... jobs offering... migrations... that makes jobs offering to diminish* (...)"
Fa: "*Salary...*"
V: "*Let's have city in the middle* (of the screen)... *we are going to have three* (variables)... *jobs offering...and migration*"
F: "*Three boxes ?*"
V: " *Yeah. After we decide who is who... we will give names to them...*" (they created 3 variables and afterwards named them)

The first model they constructed used only the default elements of LinkIt: 'smooth' variables and 'go together' links (see Figure 10.6).

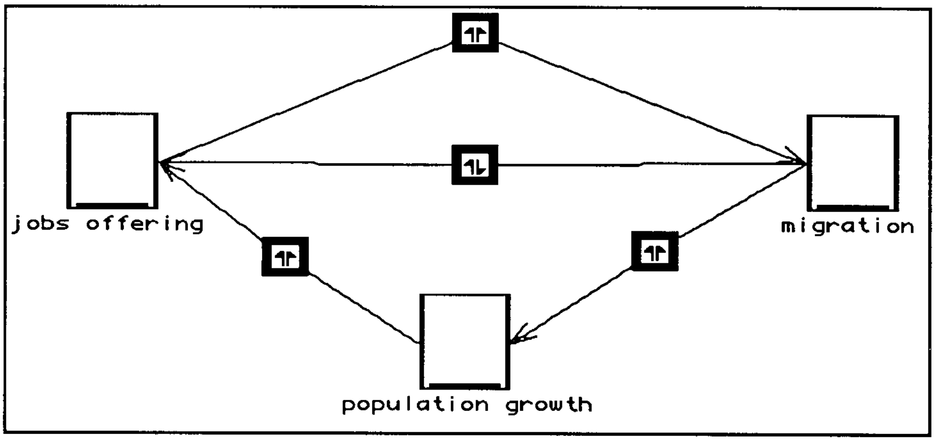


Figure 10.6: Group J: First version of the model about migration to big cities

Other things like whether a variable should be 'On/Off' or links made 'cumulative' were left for later when they moved to a second stage concerning the behaviour of variables and their links. Here they decided to delete a link between *migration* and *jobs offering* because:
V: (..) as I'm understanding, migration is together with population growth"

They also changed the type of *population growth* to 'On/Off' and made all links 'cumulative'. They ran the model many times and from what they were observing on the screen they went on to change links and properties of the variables:

F: " *Do you think that if jobs offering went to zero then migration had to go* (to zero) *as well...?*"

V: " *Not to go to zero but* (it has to) diminish" (after that they changed the effect of the link to 'weak')

During this process they were gaining confidence in what they were doing and by the end of the task they were able to defend their point of view about certain aspects of the model they constructed (see Figure 10.7), justifying it on the basis of their own experience together with what they created on the screen:

(This passage is about what the variable *population growth* was representing. At the beginning of the task, they were moving between the ideas of this variable representing the total population and the rate of growth of the population. By the end of the task, when they became more confident Vera was able to explain it. Her idea was that it became steady because there was not migration anymore. Also it was not zero because it was showing the rate of growth of the population and not the population of the city.)

F: "*That idea you mentioned...when the migration finishes, the population continues to breed, is it represented in the model ?* "

V: " *I think it is ...*"

F: " *Where it is represented ?*"

V: *In the fact that, for example... when we ran it ...the migration could go down but the population growth continued...* (...) *It is not going to diminish because that people already migrate, they are not going back* (to the rural areas)... *they are not going to come back to migration, do you understand ?* (...) *they came to the city and they are here... although jobs offering fell, they are already here .... they are breeding here, it means it is increasing the number of people in the city.* "

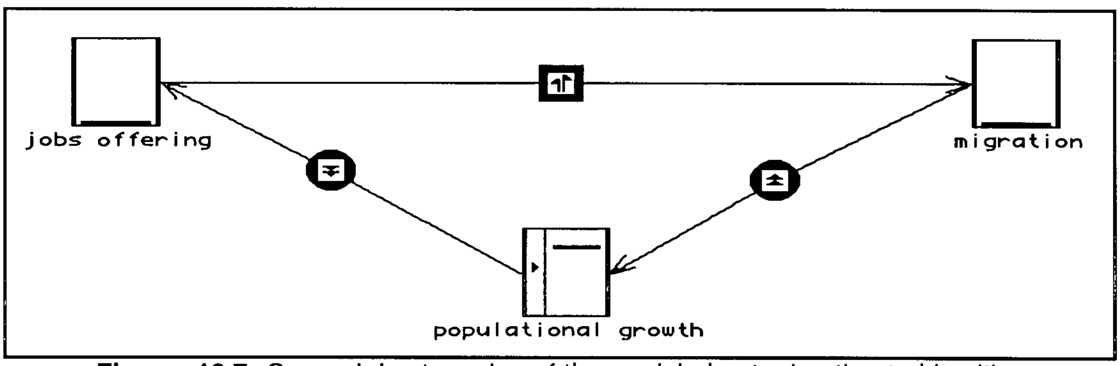


Figure 10.7: Group J: Last version of the model about migration to big cities

Before moving to the next subsections where a discussion about the progress made by the students during the last two stages of improving their models is made, it is worth mentioning that both ways of improving the models -

which I shall call "Fixing models and ideas" and "Sophistication of models and ideas" - usually happened together with the students moving from one to the other. So it could happen that while they were engaged in a process of sophisticating a model, they might find a variable not working in the way they wanted. In these cases they normally went on to "fix" this localised problem and afterwards continued with their broader goal.

The passage that follows gives a good sense of these two stages happening together and a student exploiting the possibilities of the system to achieve a quite interesting representation of the problem. It happened with Group J, when constructing a model about a "Heating system" (Task 5 activity 3).

> Since the beginning Fabricio had a picture of how a heating system works. His problem was more how to represent it within the system. But the process of testing things and reflecting on the feedback given by the system led him, in the end, to a simple but coherent model.
>
> The model he had in mind had three variables: *outside temperature, inside temperature* and *heater*. For him *outside temperature* represented how cold it is (higher the level is, colder the temperature).Therefore the reason for *outside temperature* being 'only positive values' was the fact that it would only represent negative values. Also the idea of cold was associated with temperatures equal or below zero. *Inside temperature* in its turn was changed to 'any value' to represent the idea of "*above the middle is hot and below it is cold*". *Heater* was an 'On/Off' variable that would function as a thermostat controlling the temperature of the house. He decided to trigger it when the level was above the threshold.
>
> After creating the variables and giving a name to them, he moved on to think about the links. He first created all links using the default properties of the system and only after the first simulation he thought about their properties (see Figure 10.8). After running it he said:
>
> Fa: " *The idea was that when the heater turned on, the level* (of *inside temperature*) *went in the opposite direction of outside temperature... it doesn't matter whether it went up or down... I think that one of these two* (links) *should be opposite direction* "
>
> The feedback from the system gave him the first clue about the links: one of them should be 'opposite' to make things go in the contrary direction. He decided to change the link between *outside temperature* and *inside temperature* to 'opposite direction'. Again the model did not behave in the way he wanted. Then he changed back the link between *outside temperature*

and *inside temperature* and changed the link between *heater* and *inside temperature* to 'opposite'. After running the system, he said the model was more or less functioning in the way he wanted. To match his ideas with the model he had constructed, he had to change a little his interpretation of the variables: " *I can say that from here to the top I can consider the inside temperature as cold... and to the bottom as hot... if it is in the middle it means normal...*" (see Figure 10.9).

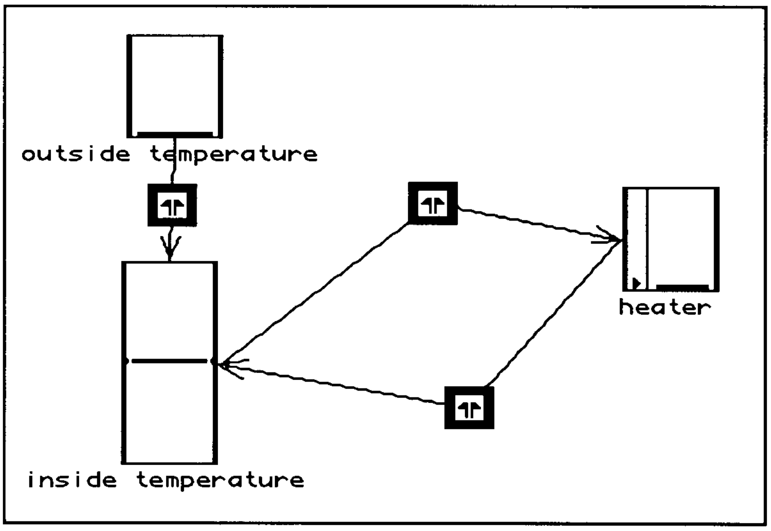


Figure 10.8: First model about the heater (Group J)

At this point it seemed he was satisfied with the model but he decided to make a last change: To change the type of the two links between *inside temperature* and *heater.* Now with all links 'cumulative' the model was again not behaving properly. This made him begin to think that the 'On/Off' variable should be triggered 'when it is below'. Then I told him that we normally think about positive values being above negative values and therefore his interpretation of *inside temperature* could cause some confusion. In order to adapt the model to this suggestion he made the link between *heater* and *inside temperature* 'same direction' and the link between *outside temperature* and *inside temperature* 'opposite'. He ran the model again and he was satisfied with what he saw (see Figure 10.10), giving the following explanation about the variables and the model:

Fa: " *The inside temperature when it is in the middle, it means it is normal, it is zero... above the middle it would be positive and downwards it would be negative temperatures.* (...) *The heater is going to vary according to inside temperature. When it is down there... very cold... the heater turns on and makes the inside temperature goes up to* (become) *hot.*"

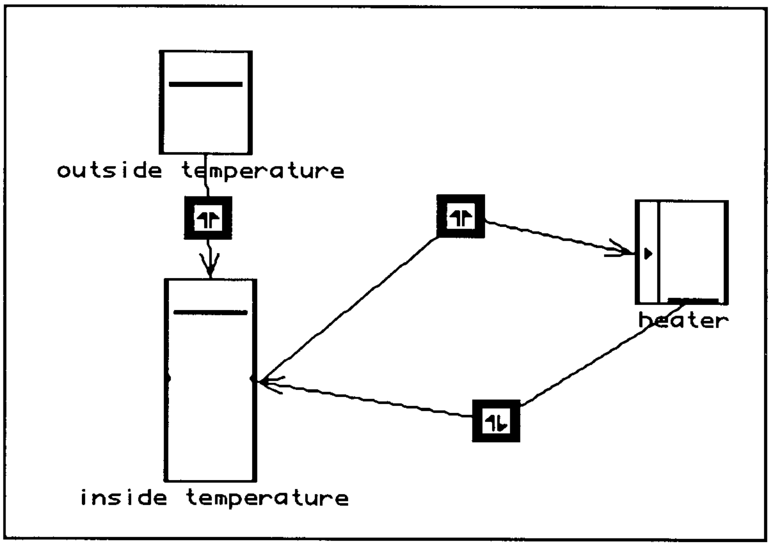


Figure 10.9: Group J: Intermediary version of the model about the heater. *Outside temperature* is representing cold temperatures only. *Inside temperature* can represent cold temperatures (above the middle) and hot temperatures (below the middle)

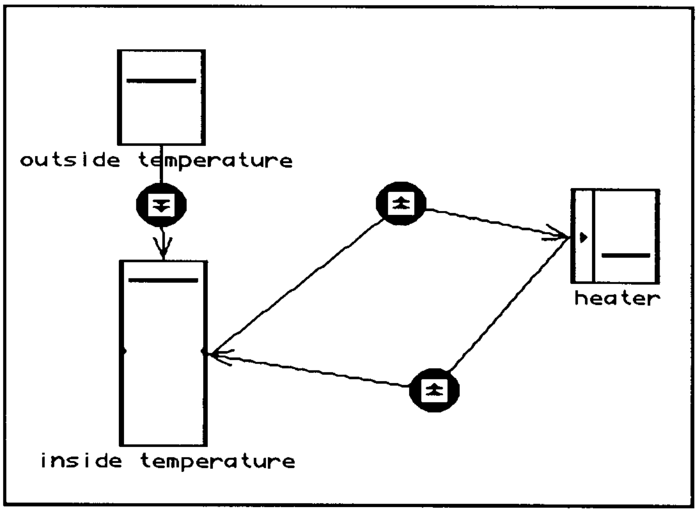


Figure 10.10: Group J: Last version of the model about the heater

10.2.3.1 Fixing Models and Ideas

Improving the models in order to "fix" some problems with the variables and their relationships took most of the time of the students when they were engaged in expressive and exploratory tasks.

The need for these improvements was normally signalled when the students observed a simulation and saw that a certain variable was not working in the way they wanted. In these situations the feedback given by the system played an important role in helping the students to elaborate their models and ideas.

The process of fixing a model varies from a simple change to the 'weight' of a link to the construction of "modelling gadgets". Below are presented different passages where the students tried to "fix" their models in different ways. It is also interesting to notice how much of their daily-life experience they employed to justify the changes made.

• <u>Changing the property of a link/variable</u>

This first passage is an example of changing the type of a link. It happened with Group I during Task 5 activity 3 (Model about "how a refrigerator works"). It is also interesting to notice here that the visual feedback of the system made them more analytical about their ideas:

> They had created a new variable called *consumption* (of energy) to measure how much energy the refrigerator is spending and connected it to *thermostat* via a 'go together-same direction' link:
>
> M (the model is running):" (...) *when the thermostat goes down (turn off) the consumption will fall ?*"
> F: " *I don't know...what do you think is going to happen ?*"
> M: " *Let's set the thermostat upwards...No this* (the link between *thermostat* and *consumption*)*here has to be ball* (cumulative)... *because consumption is increasing just a bit. It is not real.*(what is shown there).. *Consumption goes on accumulating*"
> F: " *Why in the beginning did you set it to square* ('go together') *?*"
> M: " *I put it* (like that) *just to know... to see...I did not think very much about it...I would let it run.... if it behaved well... I would leave it like that*"

This second passage is a case of changing the parameter 'when to trigger' of an 'On/Off' variable. Like the previous example, the visual feedback of the system was the main cause for the modification. It happened with Group J during Task 3 activity 7 (Introducing 'On/Off' variables):

> The example they suggested was about someone trying to resist smoking (see Figure 10.11). First *resistance* had to be triggered when it was above the level, because, for them, the idea of the level of *resistance* increasing meant that the person was having less and less resistance:

Fa: " (...) *we could work with* (a problem about a person being) *addicted...When the guy has the desire* (to smoke) *until a certain level where he doesn't resist anymore and he goes and smoke*"

However, they used a 'cumulative-opposite direction' link between them which would make *resistance* decrease. After running the model and seeing that nothing happened to *resistance* they said:

Fa: " *So we could do this: Instead of increasing it could be decreasing...*"
V: " *'ON' when it is above... 'ON' when it is below* (the trigger level) *isn't it ?*"

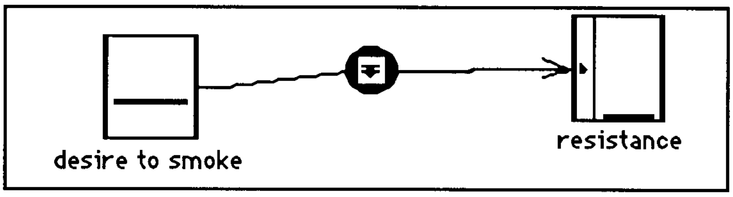


Figure 10.11: Group J: Model about resistance to smoking

The passage above suggests that the students were working with two contrary ideas: "level of resistance" and "desire to smoke". Although they did not explicitly notice this confusion, after running the model they had to adjust it to one of the two ideas.

In this third passage they employed their knowledge about the real world to suggest either a change in the strength of a link or to introduce a new variable. It happened with Group D during Task 3 activity 2 (Exploring a model about "Accidents on the road"):

When I was running the model I asked them to pay attention to the behaviour of *accidents with death* and after they said:
L: " *It seems that the total number of accidents is always connected to accidents with death, then whenever an accident happens the person is going to die. Do you understand* (it)*? You have to put something else that influences accidents*"
M: " *This accident with death. This connection* (between accidents and accidents with death) *could be weak* (...) *or to insert something that would influence this accident with death... like the quality of the rescue, how to rescue the victims*"

An interesting point is that at that time there was a big campaign on the television to give financial support to the rescue teams that work on Brazilian roads.

• <u>Changing a link and the interpretation of the model</u>

The passage below is another example where the visual feedback of the system led them to make some changes in the model which culminated in a more realistic interpretation of the problem. It happened with Group C during Task 8 (Constructing a model about "Pollution in big cities"):

> After running the model and not seeing *control of pollution by the government* becoming zero when *pollution* went below the trigger level (see Figure 10.12), they decided to change the type of the link between them to 'go together' (The argument used here was based on the expected behaviour of the variables on the screen):
>
> D: " (...) *if we put ball* ('cumulative' link) *there, pollution will increase... then the control of pollution will increase as well...but then pollution will diminish... and the control will not... then it is not going to have pollution anymore.* "
>
> They ran the model again and noticed that *pollution of the air* was not diminishing. After another simulation with new initial values they concluded:
>
> R: " *Now it arrives in a place where it is equal... the sum of cars and industries is the same as* (the value of) *control... then the pollution... is equal* (meaning not changing)..."

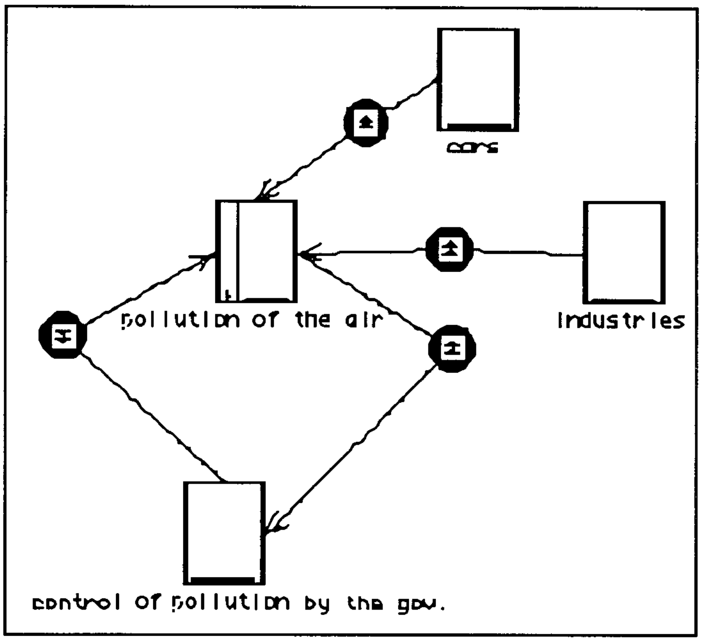


Figure 10.12: Group C: Model about pollution of the air

Later they decided to connect *control of pollution by the government* directly to *industries* and *cars* via 'cumulative-opposite' links[8]. But after running the model they did not see *pollution of the air* going down after the government began to act. They reflected a bit about the origins of the problem and came up with a new interpretation of the consequences of *control of pollution by the government:*

D: " *Now here... it changed the way the government is making the control... from now on the government is doing... stop emitting pollution... instead of diminishing the quantity of pollutants, do you understand ? Now it is realistic, it stops producing pollutants... then it is not going to have a way to pollute the air more ... that is why it* (pollution of the air) *doesn't diminish anymore* "

• Employing "modelling gadgets"

Another way of trying to "fix" a problem in their models was by creating some "modelling gadgets". These mechanisms were normally employed after students failed to force a certain behaviour on a variable (or on a group of variables) using the properties available. An example happened with Group D during Task 8 and is shown under the second passage in the topic: Constructing "modelling gadgets". - Section 9.2.1.2. There they did not like the behaviour of the variables *max. use of buses* and *use of cars* and decided to create two new variables to restrict their behaviour.

8 This is a case where the stages of Sophistication of a model and fixing it are mixed.

10.2.3.2 Sophistication of Models and Ideas

The transcriptions of the interviews are full of evidence of students' ideas becoming more robust as their work in a certain task progresses. This evidence can be shown by what they said about a particular problem and by the models they constructed/modified.

In the case of exploratory tasks the increasing complexity of their ideas can mainly be shown by what they said about a problem at the beginning and later, after interacting with the model presented to them. Particularly during Task 5 activity 3 (working with the model about "refrigerator") I asked the students, at different moments, to describe how a refrigerator (or air-conditioning in the case of Group A) works. The following two passages are about two different groups talking about it.

> Before Group A loaded the model, I asked them how they thought an air conditioning works and they said:
>
> C: " (...) *it cools the air that circulates in the room* ".
>
> Later (after loading and running the model) they gave a more complete description based on what they were observing on the screen:
>
> C: "*When the room temperature becomes high... then...the air conditioning turns on and diminishes the temperature and when it increases again...*"
> L: " *When the temperature gets to a certain height like this...hot...then the air turns on...decreases the temperature of the room and increases the difference of the temperatures*"
> C: " *Then it turns off the air,* (the temperature) *increases again, turn on the air again, OK ?* "
>
> During the same Task Group D reacted in an interesting way. In the beginning, when asked about the refrigerator, Lucio talked much more than Marcio and both used in their speech important elements for the problem such as consumption of energy, temperature and thermostat without connecting these ideas appropriately. When particularly asked about how a thermostat works, Marcio said he had no idea. Later, after loading the model and running it, he was able to explain the problem using in his description many of the variables existing in the model and the dynamics of the system:

M: "*Yeah, the temperature* (of the refrigerator) *has to be kept low, if it arrives at a very low level, the motor* (of the refrigerator) *will begin to work less and it* (the temperature) *will begin to increase and when it arrives in a certain level the engine has to turn on again to diminish the temperature* "

He was also able to describe the *thermostat* in terms of its role in the model:

M: " *After it* (the refrigerator) *arrives at a certain temperature, it* (the *thermostat*) *has to start the engine to diminish the temperature* "

During expressive tasks the increasing sophistication in students' ideas can also be observed in an increase in the complexity of the models constructed. This sophistication comes, in most of the cases, after making the model behave in an appropriate way (after fixing the problems concerning the behaviour of the variables on the screen) and is normally implemented by using a combination of the two forms below:

- By including new ideas (variables) in their models;
- By moving from a "snapshot view" to a more "dynamic view" of the problem.

The passages below are examples where these two forms of making models and ideas more sophisticated were employed. They are presented together with tables that give some indices about the models constructed by the students[9]. Basically the indices are as follows:

(1) Simulation - Explains which simulation the other indices refer to. In that way, the first simulation refers to the first model the students constructed before attempting to run it. The second simulation refers to the model existing by the time of the second simulation and so on.

(2) Var. Tot. - Total number of variables existing in the model

(3) On/Off Var - Number of 'On/Off' variables existent in the model

(4) Cumulat. Links - Number of cumulative links existing in the model

(5) Linkg - Linkage: number of links/ total number of variables

(6) Interdep - Interdependency: number of dependent variables/ total number of variables

(7) No. of Fdbk loops - Number of feedback loops

(8) Remarks - Remarks about the structure of the model.

[9] The idea of these tables was drawn from the work with IQON where a simplified version was used (Bliss, et al., 1992b).

The tables do not contain a description of all models. They only have entries for the models in which there was a change in the number/name/type of variables and/or in the number/type of links and/or in the number of feedback loops.

• Including new ideas (variables)

The process of including new ideas in their models could be done either by changing the name of variables already existing or by creating new variables.

When they changed the name of a variable as a resource to implement an improvement in their models, they could either go from a general name to something more specific (which implied the creation of new variables to represent other particulars of the problem being modelled) or go to something more general (simplification of ideas). Below are given examples of both cases.

> In this passage, Group H was creating a model about pollution in a big city (Task 8)[10]. Fabricio first created the model shown in Figure 9.14.
> Until the eighth simulation he was more involved in representing the main ideas about the problem and in making the model work in the "correct" way (Table 10.1 shows that between the first and eighth simulations there was no change in the number and types of variables, number and types of links or creation of feedback loops)
> By the time of the ninth simulation there was a change in his thoughts and he began a process of making his model more sophisticated by giving more detail to some ideas already there:
>
> Fa: " *Something is going wrong with this model because the street cleaners they didn't go* (to clean) *the waste of the industries...of the air, of the cars...So it would have to have a box for pollution of the air, pollution of the water and pollution of the ground* (...)"
>
> He then changed the name *pollution* to *total pollution* and created new variables to represent specific types of pollution: *pollution of the air*, *pollution of the ground* and *pollution of the water* . Between the ninth and fourteenth simulation he kept adjusting his model and he then decided to create a new variable - *sewage* (the idea was to represent the population polluting the water)- because the pollution of the water remained very low during the simulations . By the sixteenth simulation he decided to make the

[10] Fabricio was working alone during this Task.

links 'cumulative' in order to use the model to represent something evolving with time :

F: " *How are you interpreting this problem ?*"
Fa: " *Like this... if you have cars polluting X every day, industries polluting X every day then pollution of the air will be 2*X*"
F: " (...) *If I want to think about this problem over time... I have a city...with a lot of cars not too much industry...the street cleaners don't work very well...*"
Fa: " *It would be necessary to change all boxes to time... to time relationships* ('cumulative' links)"

After that he performed some more simulations and we began to discuss the pollution of the air (it was very big in his model). He first thought that there was no way to diminish it and a bit after he said:

Fa: (..) *Winds !... The wind carries the pollution... Rio de Janeiro is a city close to the sea... it has a lot of wind...the pollution will diminish*"

He then created a new variable called *winds* and connected it to *pollution of the air* (see Table 10.1 - twenty fifth simulation and Figure 10.13)

SIMULATION	Var. Tot.	ON/OFF Var.	Cumulat Links	Linkg	Interdep	No. of Fdbk loops	Remarks
First	6	0	0	5/6	1/6	0	star/medium
Ninth	9	0	0	9/9	4/9	0	star/big
Tenth	9	0	0	9/9	4/9	0	star/big
Fourteenth	10	0	0	10/10	4/10	0	star/big
Sixteenth	10	0	10	10/10	4/10	0	star/big
Twenty fifth	11	0	11	11/11	4/11	0	star/big

Table 10.1: Group H working with the model about pollution in big cities (Task 8)

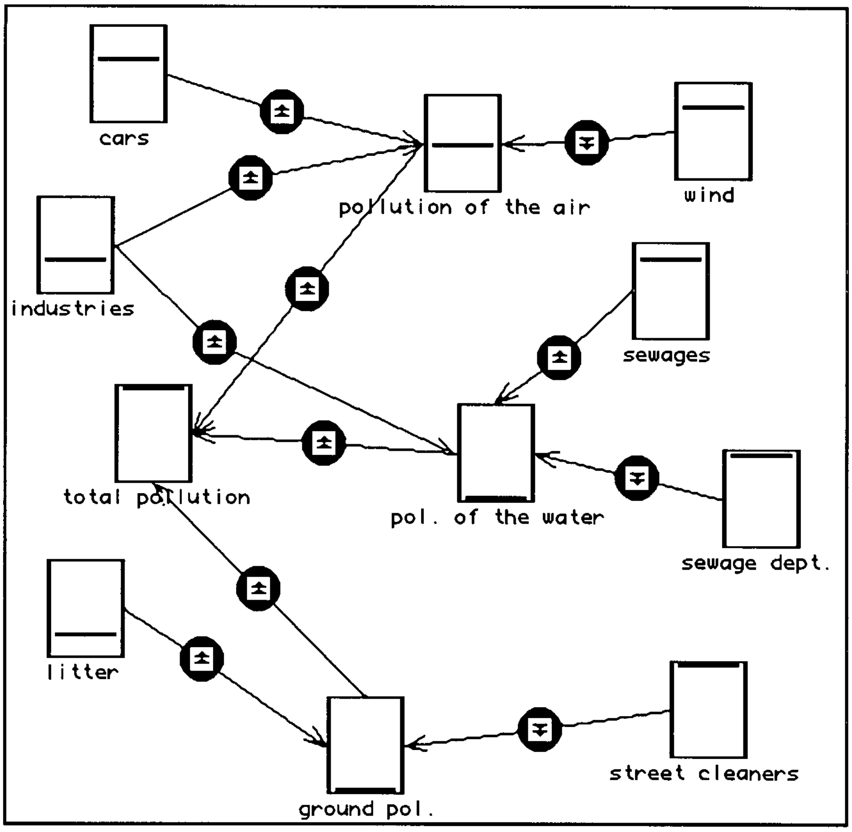


Figure 10.13: Group H: Last model about pollution: Three variables representing different types of pollution

The next passage happened with Group A during the construction of the model about Migration to big cities (Task 4).

First they suggested the model shown in Figure 10.14.

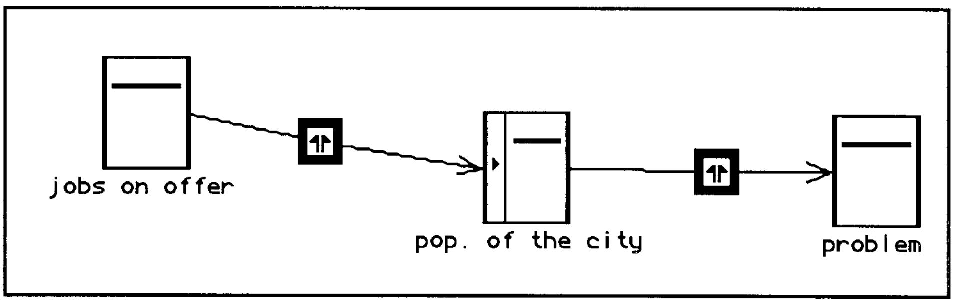


Figure 10.14: Group A: Model about migration to a big city. Later *jobs on offer* became *attractive* (attractive factors)

After running it and not being satisfied with the behaviour of *population of the city* (it was increasing very fast) they decided to change its ingoing link to 'cumulative' (second simulation). Later, by the fourth simulation (see Table 10.2) they began to realise that *jobs on offer* was not the only cause for making people move. They then decided to change its name to *attractive* in order to incorporate other important factors that cause population to increase:

C: "*We could change jobs on offer to attractive* (attractive factors)"
F: "*... to attractive. It is OK. Why do you think...*"
C: " *Because it is not only jobs on offer... it has...*"
L: " *Better life conditions...(..)*"

SIMULATION	Var. Tot.	ON/OFF Var.	Cumulat. Links	Linkg	Interdep	No. of Fdbk loops	Remarks
First	3	1	0	2/3	2/3	0	chain/small
Second	3	1	1	2/3	2/3	0	chain/small
Fourth	4	2	3	4/4	3/4	1	chain/medium
Fifth	4	2	3	4/4	3/4	1	chain/medium
Sixth	4	2	3	4/4	3/4	1	chain/medium

Table 10.2: Group A working with the model about migration to big cities (Task 4)[11]

After this (and still between the third and fourth simulation) we started thinking about the problems of the population moving to big cities and what could be done to alleviate these problems. Their first idea was to offer better jobs and pay better salaries, which, according to them, could be created with a 'same direction' link from *problems* to *attracting factors* . But they soon realised that it would end up in more *problems* (it is a positive feedback loop) and decided to think about something that would affect *problems* only. Their suggestion was to have some kind of social service which would construct houses and shelters for children and homeless. The general name they gave to this idea was *improvements.* They then suggested making *problems* 'On/Off' and connected it to *improvements* via a negative feedback loop:

L: "*When problems increase, improvements increase as well and when improvements increase the improvements diminish the problems*"

After that they did some more simulations (the last two simulations shown in Table 10.2) to adjust the 'weight' of the links involved in the feedback loop in order to make the model behave in the way they wanted.

[11] This table was created from the transcription of the tapes only because there was not a log file for this Group.

• <u>By moving from a "snapshot view" to a more "dynamic view" of the problem</u>

The students also demonstrated an increase of sophistication in their models and ideas by moving from a "snapshot view" to a more "dynamic view" of the problem. Normally in these situations the students implemented feedback loops with 'On/Off' variables or by changing some links to 'cumulative' in order to make the models evolve with time.

The first example below is a case where the students employed feedback loops and 'On/Off' variables. It happened with Group E during Task 6 (Constructing a model about "Diet and healthy life"):

> After reading the text, Diego decided to represent in his model the main ideas about diet and health with the implicit goal of achieving a good health for a certain person . He constructed his model bit by bit and by the third simulation he had the model shown in Figure 10.15.

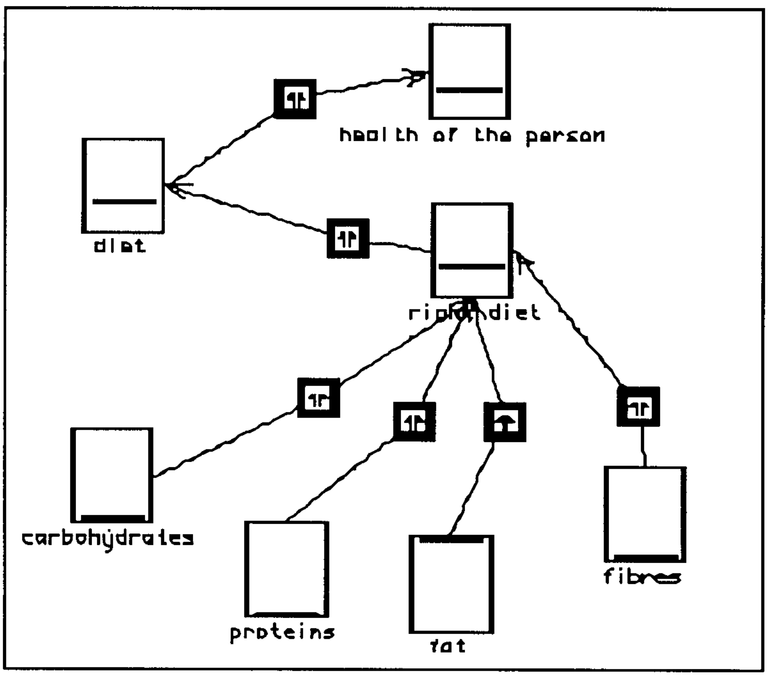


Figure 10.15: Group E: First model about diet and healthy life

> After he had tried some simulations and made small changes to make the model behave in the way he wanted (between the third and twelfth simulation - See Table 10.3), I decided to ask him what happens when health becomes bad. He promptly suggested a new improvement in the model making *health* 'On/Off' and creating a feedback mechanism to improve the *quality of the diet* (see Figure 10.16 and Table 10.3 between thirteenth and eighteenth simulations):

F: *"OK. and what about the case when the health of the person become bad...?"*

D: " *It could change this box here...health to On/Off... It turns on if above... (...) But I should connect health of the person with all carbohydrates... proteins... fat and fibres...(...) When his health is bad he has to eat more fibres, more fat, more proteins and more carbohydrates...*"

SIMULATION	Var. Tot.	ON/OFF Var.	Cumulat Links	Linkg	Interdep	No. of Fdbk loops	Remarks
First	2	0	0	1/2	1/2	0	chain/small
Second	3	0	0	2/3	2/3	0	chain/small
Third	7	0	0	6/7	3/7	0	star-chain/big
twelfth	7	0	0	6/7	3/7	0	star-chain/big
thirteenth	7	1	0	6/7	3/7	0	star-chain/big
fourteenth	7	1	0	8/7	5/7	2	star-chain/big/high linkg
fifteenth	7	1	0	10/7	7/7	4	star-chain/big/high linkg
Seventeenth	7	1	0	10/7	7/7	4	star-chain/big/high linkg
Eighteenth	7	1	0	10/7	7/7	4	star-chain/big/high linkg

Table 10.3: Group E working with the model about diet and healthy life (Task 6)

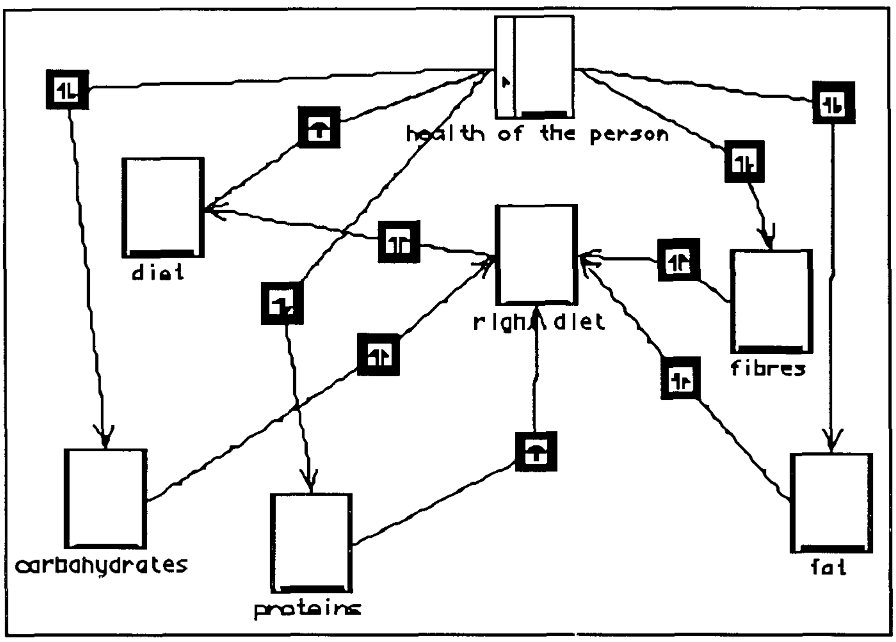


Figure 10.16: Group E: Sophisticated model about diet and healthy life

The second example happened with Group C during the same Task. Here they employed feedback loops and changes in the type of some links to move from an initial idea where they wanted to represent "how healthy a certain person is" to a more dynamic view where it was possible to discuss about the mechanisms that control someone's health.

In the beginning of the task their aim seemed to be to create a model to show how much good health someone has according to how much he/she eats from each group of food (see Table 10.4 - first simulation and Figure 10.17):

F: " (...) *what do you want to show there ?* "
R: " *You eating correctly, you are going to have a good health...but if you consume more than it is needed...then ... your health is going to diminish...*"

The threshold of the variables represented the correct amount of each kind of food someone can have before *health* begins to fall (the height of the threshold would represent the correct percentage of each group).

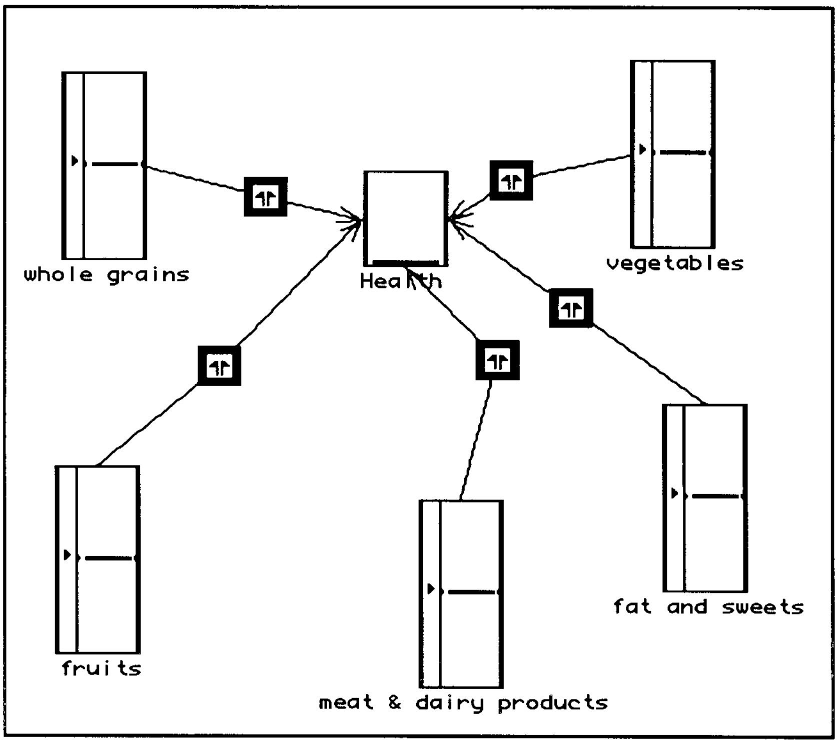


Figure 10.17: Group C: First model about health and diet

SIMULATION	Var. Tot.	ON/OFF Var.	Cumulat Links	Linkg	Interdep	No. of Fdbk loops	Remarks
First	6	5	0	5/6	1/6	0	big/star
Second	7	5	0	7/7	2/7	1	big/star
Third	7	5	1	7/7	2/7	1	big/star
Fourth	7	5	6	7/7	2/7	1	big/star
Fifth	8	5	7	7/8	3/8	2	big/star

Table 10.4: Group C working with the model about diet and healthy life (Task 6)

When they were satisfied with the model, they started thinking about it as something that could represent the mechanism of keeping a person healthy (see Table 10.4 from the second simulation on):

R: " *Maybe when it* (*health*) goes below (a certain value), *the person would become alert...would eat more grains ...*"
D: " *Diminish the fat...* (...)"

To implement this new idea Rogerio suggested that the links should be in both ways (use of feedback loops) because "*when the health goes down it will make the person to eat less fat*". They then decided to represent a situation where while a person was eating a certain amount of a certain food below a established level, his/her health would increase. When this person begins to eat above this level his/her health begins to decrease. To implement it they changed the links departing from the 'On/Off' variables to 'cumulative' (to implement the idea of health changing with time) and created four new 'smooth' variables that would work in conjunction with each one of the four 'On/Off' variables. In order to try their idea they first tested with only one variable - *meat and dairy products* - putting all the other three variables to sleep (see Table 10.4 - third simulation). The mechanism they created had the form shown in Figure 10.18 .

They explained it as follows:

D: " *It has to be opposite* (the sign between meat and health)... *when it* (meat and dairy prod.) *turns on it has to diminish health...when health increases the guy would be very happy and would eat a lot of meat and dairy products...and it would increase meat and dairy products* (...) *and it would arrive at the threshold and it would make this one* (health) *to decrease* (...)"

After being satisfied with it they decided to "copy" this mechanism to all other variables (see Table 10.4 - fourth simulation on).

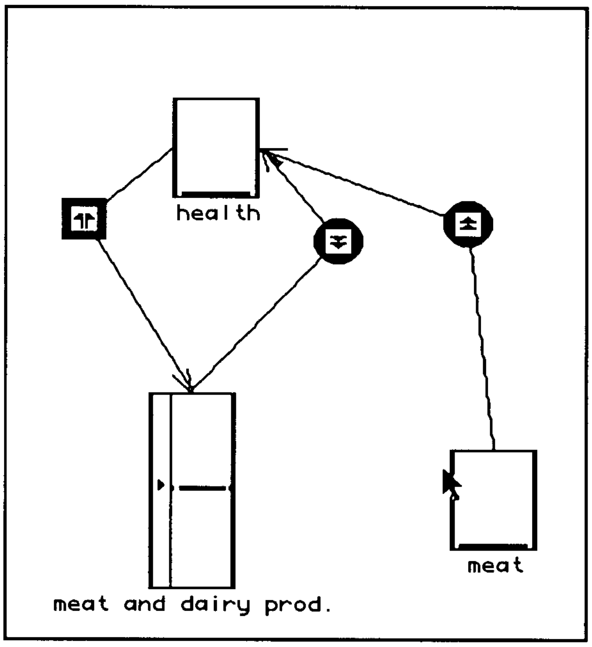


Figure 10.18: Group C: Mechanism created to implement the idea of health changing with time and different consumption of meat and dairy products

10.2.4 Recognising Analogous Model Structures

Finding similarities between models is an important step towards understanding categorisation and thinking about problems using higher levels of abstraction.

When working with LinkIt students were able to perceive both the underlying way of working of components of the system and similarities in the underlying structure of different models. Actually Task 7 was specially designed to observe students finding a problem that fits a model structure presented to them. An interesting case where the students were paying attention to the whole model structure happened with Group D (during Task 7) when presented with the second empty model (variables without names and links):

The second model I showed them had the general form

XXX -- (**Cum, same**) ---->

---- **Add** --> ZZZ

YYY -- (**Cum, oppos**) -->

The kind of model I had in mind was

Rate of birth -- (**Cum, same**) -->

---- **Add** --> *Population*

Mortality -- (**Cum, oppos**) --->

After analysing the properties of the model they suggested the following model

Food intake -- (**Cum, same**) --->

----- **Add** --> *Weight*

Exercises -- (**Cum, oppos**) -->

I told them it was right and asked for another example that could fit in this situation. They suggested a problem about the population of a city where there were people coming into and going out of the city.

I then decided to explore further their ideas about the first solution they gave. So I asked them to think about the problem of gaining and losing weight in real life and how it could be represented in the model. The solution they gave was:

M: " *To set weight to On/Off...When it arrives at a certain level he could eat less or do more exercises*"

F: " *Could you show it for me ?*"

...

M: "*Weight becomes On/Off...*"

L: " *We have to put a connection from weight to exercises...*"

F: "*In case I want to do something about diet...*"

M: " (you) *should make* (a connection) *between weight and food intake*"

They not only promptly knew how to solve the problem I proposed but they also knew that depending on the problem I wanted to focus on the model should be implemented in a different way although the basic structure would not change.

The students also recognised similar underlying structures when engaged in other tasks. An example happened with Group A during Task 3 activity 6 (Completing some models):

They were completing the model about gaining and losing weight and I asked them to compare this problem with the one about the "waiter" and the "bath tub" and Carmen came up with an explanation:

C: " (...) *this problem could be* (seen) *like this: energy spent would be the drain and how much you eat everyday would be the tap. One makes it to fill and the other makes it empty* "
L: " *Yeah. That is right* "

Another way of recognising analogous structures was by explaining a situation using a similar one to support their ideas. An example happened with Group C during Task 5 activity 3 (Model about "refrigerator and heater"). When asked about how a refrigerator would work on a very hot day they made a comparison with air-conditioning saying:

R: " *It is like air-conditioning, it will take more time to cool the* (inside) *air, the refrigerator is going to be hotter... the process will take longer*"

Also during the work with LinkIt, when they forgot a name of a property or the type of link they recalled them by talking about previous problems in which they were employed. For example, a frequent way of referring to a 'cumulative' link was by recalling the first problem where it was introduced (problem about the bath tub). In the same way they used the problem about clouds and sun shining to talk about combining links with 'multiplication' (to have the effect of 'inverse').

10.2.5 Relating Models and the Real World

During the work with LinkIt the students employed quite a lot of their knowledge about the real world in at least two different ways:

- By talking about (new) related problems;
- By using their knowledge about the real world to justify actions/behaviour with the models being constructed/explored.

Let us now discuss each of these in turn.

• Talking about (new) related problems

Sometimes the work with the models led them to bring to our meeting some issues they had been discussing in their classrooms. An example happened with Group J during Task 6:

At the end of the task they began to use their model to calculate how good or bad their health was. Probably because they noticed that according to their model, fat had a bad effect on someone's health, they said:

V: " *In the Biology class yesterday we were talking about why men have more heart disease* (than women)... *we were saying that men have a stronger tendency to have their veins blocked... because the hormones that women have, because of their menstruation and the oestrogen, that makes fat accumulate under the skin causing cellulite...but men because they don't have oestrogen* (they) *don't accumulate it*" (the lack of oestrogen is a cause of heart disease in men but it is also responsible for cellulite in women)

• <u>Using their knowledge about the real world to justify actions/behaviour with the models being constructed/explored</u>

The students used their knowledge about the real world in many different situations to justify their actions when working with the models. The first example below is about creating a justification for what they were observing on the screen:

Group A was working with the model about the "eye-pupil" (Task 5 activity 1). After doing some tests, they accepted the answer given by the simulations saying:

C: " *It* (the eye-pupil size) *diminishes in order not to allow too much light to enter the eye otherwise it could harm the sight a lot* ."

The next passage is an example of using reality as a reason for a fault in the behaviour of a model:

Group D was constructing a model about a heating system (Task 5 activity 3). After running the model and not seeing anything happen, they changed the type and direction of the link between *indoor temperature* and *thermostat* to gradual with same direction (see Figure 10.19). The reason they gave for this change was simply "*because it* (the model) *has few gradual connections*". They ran the model again and saw that after the *heater* turned on it never turned off again. They suggested that it was because "*the summer had arrived* " and continued to consider how to fix the system.

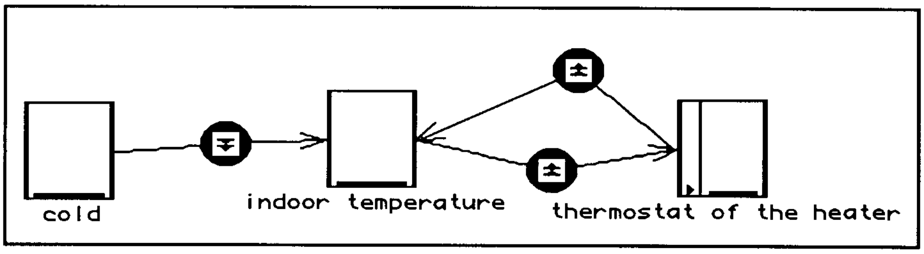


Figure 10.19: Group D used a reason from the real world for not seeing the thermostat going on and off

The next passage can be seen as an example of using a situation in the real world to justify a particular behaviour of the model:

> At the end of Task 3 activity 2 (Explore a simple model about "accidents on the road") with Group G, I asked them how they could use the model to give an explanation about accidents on the Brazilian roads (they had previously agreed that the model served to represent the reality of Brazilian roads). Because they set some initial conditions and ran the model only once, I decided to repeat their steps, saying out loud what I saw:
>
> Fa: " (...) *I will set...there bad quality of the roads...lets see what happens to total number of accidents...run it...Then I say ... it's OK... I don't know whether the bad quality of the roads contributes to cause accidents..*"
> A: " *Many cars when they see a hole* (big cracks) *in the middle of the road, they go around it... and when another car is coming... they crash..*"

Another possible interpretation here is that they did not understand my question and thought that I did not understand what was happening on the screen. So, in order to make it more real (understandable), they decided to give a particular (and concrete) example that could happen in the real world and which was represented by what we were seeing on the screen.

This last passage is an example of using their knowledge about the world to implement some changes in their model:

> Group J was working with the model about "Diet and good health" (Task 6) and gave some justifications for changing the weight of some links in their model:
>
> F: " *I would like to understand it* (...) *what are you using to change the weight of the relations ?* (...)"
> V: " *I am using my own concepts* (ideas) *about diet* (..)"

F: " *Can you explain these concepts ?*"
(...)
F: " *Meat and dairy products, aren't they very important ?*"
V: "*Yes* (but) *you can live without them...vegetarians... The life expectancy in India is bigger than in Brazil* (she is assuming that in India people don't eat as much meat as in Brazil) .... *wholemeal products, it is not very common* (to see people eating them) *it is something important...many people have problems with the digestive system... their bowels don't work well*"

An important aspect of these reactions is that they can be seen as evidence that students, in some way, are giving meaning to the modelling process itself and reflecting upon how things work in the real world.

10.2.6 Making Predictions and Testing Them

In most cases, the students used their model to present an idea they had about a certain domain. The act of formulating and testing predictions seemed to be something they did mentally. Once the model was constructed, they expected it to show the "right" answer about a certain situation. As a consequence, to design detailed tests and make exhaustive simulations becomes unnecessary. An example to support this argument happened with Group J during Task 5 activity 2 (Working with the model about the "Eye-pupil"):

After being asked about what happens to the eye-pupil on a sunny day they proceded to do only one simulation and after gave an explanation of it:

F: " *What happens to the eye-pupil of a human being when he is at the beach on a very sunny day ?*"
M (after trying one simulation with a high level of *amount of light* and medium of *comfortable level of light*): " *Keep increasing the amount of light to the maximum...on a sunny day, the beach...you're going to pay attention to this level of comfortable level of light... when you put it to run... I mean, with this small aperture of the eye-pupil... the light that enters the eye became equal to comfortable level* (of light). *If you open the eye-pupil more , the light in the eyes would become bigger and you are going to have an excess of light. If you close the eye-pupil more , you would have less than comfortable level of light, in order words a lack of light* (inside the eyes). *In this way the system is, it is in equilibrium then with this small aperture of the eye-pupil...*"

Marcio gave a complete description of the system and also made assumptions about it working in other different situations without the need of new simulations. He certainly had to run the model mentally to arrive at his conclusions about the model working under different conditions from the ones he used in his simulation.

Other groups, during the same activity, behaved in more or less the same way: they made a prediction about what was going to happen and went to the model and tried only one test to confirm their ideas. When the test did not confirm their predictions they assumed that the model was "wrong".

An interesting fact was that later in Task 5 activity 2, when the groups had to predict and devise a test to show what happens when someone goes in and out of a tunnel, they devised a test that included two simulations. One to show a person entering the tunnel and another to represent the person coming out of it. Here again this behaviour can be seen as reinforcing the argument that the students do not make exhaustive tests precisely because, for them, the models are meant to represent "right" ideas. The idea of creating more than one simulation is because the prediction they had to make (and the question put to them) implied a certain event happening under two different situations: inside and outside the tunnel. So the test reflected exactly this characteristic of their predictions.

10.3 CONCLUSIONS

In most cases the students used their models to represent/demonstrate a certain idea they had in mind. Also, when working with the tasks proposed, they employed a lot of their experience of the real world to justify their actions and the behaviour of the models being explored/constructed.

The fact that they did not use the models to formulate and test hypotheses can be attributed to the nature of the tasks presented to them. In most cases these tasks asked them to construct/explore a model in order to represent/discuss their ideas about a certain problem. Actually when these tasks challenged them to make predictions and devise some experiments to try out those predictions, the students were able to do so.

Nevertheless, more important than that is the fact that the students, when interacting with the system, had a chance to make more concrete their ideas about a certain domain. In some cases, when reflecting upon what was on the screen, they were able to propose and implement some sophistication in their models. These moments can be seen as evidence that some learning took place, permitting the students to arrive at a more complex level of articulation of their ideas.

The process of creating a model consisted basically of three stages. First the students put on the screen a list of variables and links using mainly the default parameters of these objects. Next they went on to test this first "draft" doing some simulations with it. The visual feedback given by the system played an important role in this stage, challenging the students and making them reflect upon their ideas. This reflection and the process of "fixing" the model, sometimes, led them to a third stage during which they made their models more sophisticated. Normally these sophistications were implemented by means of the inclusion of new structures in their models and changes in the view of the model from a "snapshot" perspective of the problem to something more dynamic that evolves with time.

PART IV

CONCLUSIONS AND FUTURE WORK

Chapter 11:

CONCLUSIONS AND FUTURE WORK

11.1 INTRODUCTION

The first two sections of this chapter summarise the thesis and consider its contributions. These are followed by sections that discuss its limitations and make suggestions for future work.

11.2 SUMMARY OF WORK

11.2.1 Motivation and General Ideas

The importance of computational modelling in education (and particularly in science) has become increasingly recognised by educators (see Section 1.5). Recommendations about its use are becoming more prominent and can be found both in England (National Curriculum) and in other countries (e.g., the recent Project 2061 curriculum reforms in the USA). The problem is that most of the current modelling systems are beyond primary/secondary students' abilities mainly because they require too much prior knowledge and mathematical skills (see Sections 1.5.1 and 1.6). This work moved a step ahead in this direction by proposing, designing, developing and testing a semi-quantitative modelling tool called LinkIt.

The challenge here was to propose an environment where the students could discuss models and the process of modelling by using their prior knowledge and their natural way of expressing their ideas about "real world" phenomena. Work in the field of Qualitative Reasoning and the Systems Dynamic approach provided the theoretical framework for the development of such a tool (see Chapters 2 and 3).

LinkIt provides the user with a set of building blocks from which he/she can choose and combine the appropriate objects and functions in order to create their models. The internal inferential engine of the system (which is always hidden from the user) is capable of recognising and running this structure, giving in real time on the screen visual feedback of the model evolving over time.

These building blocks are at the same time powerful enough and sufficiently mathematically consistent to permit the user to work with a wide variety of educationally interesting situations, yet sufficiently basic to make them engage in a metacognitive process about modelling (i.e., reflect about what can be done and how) without the burden of taking a long time to learn how to operate them (Chapter 4 describes a first version of the system - LinkIt I - and Chapter 6 its improved version called LinkIt II).

11.2.2 Implementations and Use of the Software

The design of LinkIt was based on two other modelling systems: STELLA (Structural Thinking Experimental Learning Laboratory with Animation) developed by High Performance Systems (STELLA is presented and discussed in Section 2.5); and IQON (Interactive Quantities Omitting Numbers) developed at the University of London (IQON is presented and discussed in Section 2.6).

The starting point for its implementation was the need for reprogramming IQON using a more "conventional" computer language in order to permit it to be more easily modified and to use less computer resources (IQON was developed using the SMALLTALK language which needs a lot of computer resources and nowadays is not very much used). The decision was to use the cT language mainly because of its facilities to implement graphical interfaces and the possibility of moving the software developed to other computer platforms.

During the process of reprogramming IQON in the cT environment, a completely new interface was proposed and implemented (at this time the STELLA interface and its modelling possibilities were taken into account in the design of the new system) and new features were also incorporated onto the system. This new version of the software was then called LinkIt I (see Chapter 4 for a description of LinkIt I).

A Preliminary study was designed aiming mainly to investigate aspects of the interface and new functionality of the system. Although some learning issues were also considered in this study, they had less importance in comparison to the design aspects mentioned before (see Chapter 5 for the description and analysis of the Preliminary study).

The Preliminary study involved seven pairs of students aged 13 to 18 years old from a state school in Rio de Janeiro, Brazil. Each group worked through a set of exploratory and expressive tasks already prepared for about 3.5 hours divided into two meetings.

The analysis of the data collected led to a rethinking of the conceptual model of the system. A new interface and changes in the properties of the objects of the system were discussed and implemented, resulting in a new version of the software: LinkIt II (see Chapter 6 for a description of LinkIt II).

During the programming of LinkIt II a further literature review concerning learning issues was conducted and a second study began to be elaborated. At this time the idea was to set up an investigation about students' success and failure with the new version of the software - LinkIt II - paying attention to the ease of use and 'learnability' of the system and to how they explored and externalised their ideas when using it. This study therefore posed two demanding tasks for the students:

(1) understanding, use and manipulation of the software;
(2) thinking and learning with the software.

The main guiding research question was then elaborated and unfolded into two subquestions which were closely related to demands (1) and (2) presented above (see Figure 11.1).

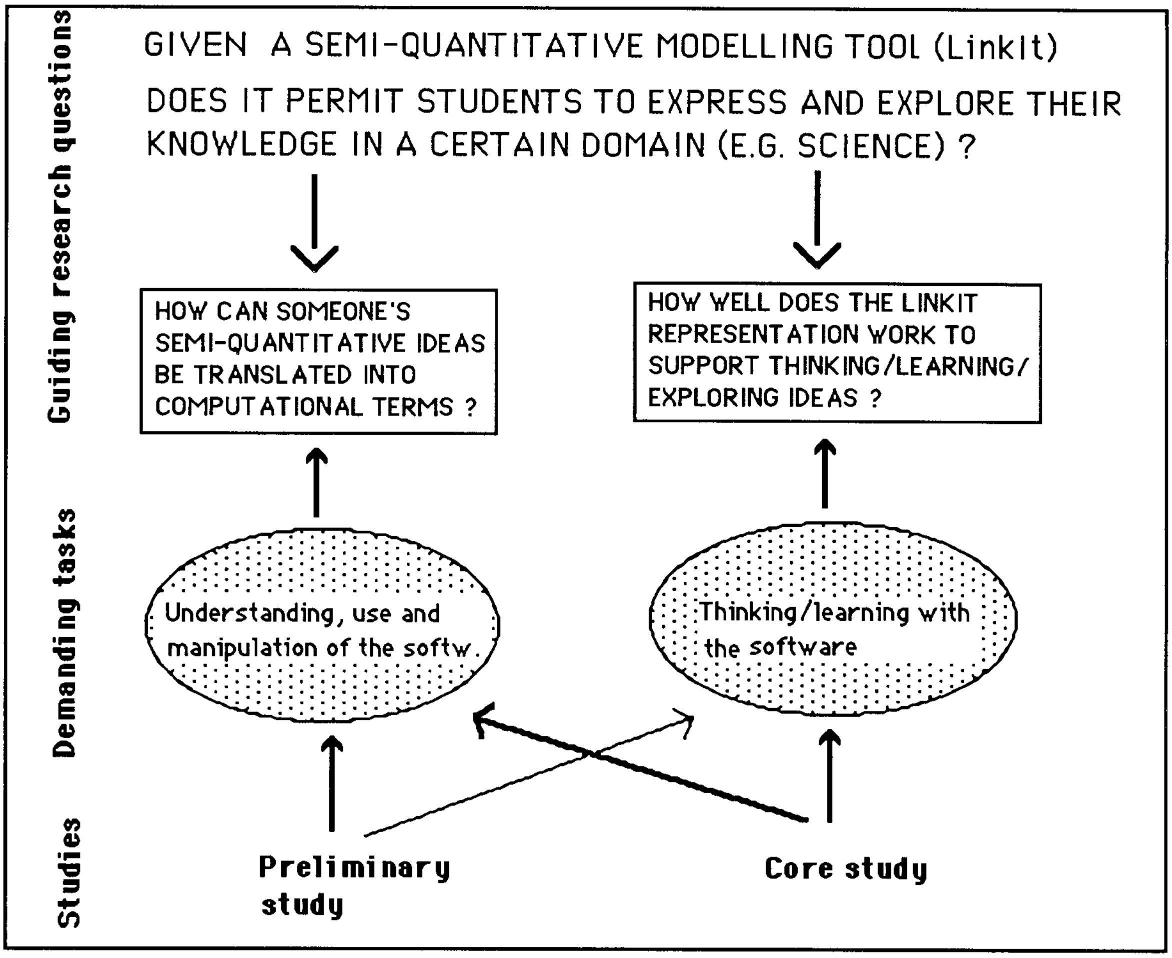


Figure 11.1: Main research questions and design of the studies

These two domains of investigation led the analysis to be divided along two dimensions, presented in different chapters. Chapter 9 discusses students' understanding, use and manipulation of the software and Chapter 10 is mainly concerned with students' thinking and learning with LinkIt.

The Core study was designed using a similar structure to that of the Preliminary study (see Chapters 7 and 8). However, at this time the number of meetings and the number of tasks had to be expanded to permit a deeper level of interaction between the students and the software. Nine groups participated in the experiment (although only eight were used in the analysis), working with a set of tasks divided into four meetings (7 to 8 hours per group).

11.3 CONTRIBUTIONS

As the work described in this thesis relates to the design, implementation and use by young students of a semi-quantitative modelling tool, it represents as a whole a contribution in the field of computer modelling in education. The practical implications of the work relate to the availability of the software developed, the design of its user-system interface and inferential engine and expectations about teenage students using modelling systems. Also the format of the studies and their analysis may be useful to other researches in the field. A summary of the main contributions of the work together with their explanations are presented in the subsections that follow.

11.3.1 Availability of LinkIt II

LinkIt II is available for both Macintosh and IBM compatible computers in English and Portuguese versions.

The Macintosh version requires at least 2 MB of RAM memory, System 7.0 and 1 floppy disk drive.

The IBM version only runs under DOS and requires at least 2 MB of RAM memory and 1 floppy disk drive.

11.3.2 Design of the System

This research has extended the work in computer modelling in education and software design by proposing and developing a semi-quantitative modelling system drawing ideas from Cognitive Science (Qualitative Reasoning and Neural Networks), Systems Dynamics and Human Computer Interface (the relevant ideas of these fields of research concerning the design/use of the software developed here are presented in Chapters 1, 2, 3, 4 and 6).

The innovative aspects of the software developed can be summarised as:

• A graphical user-system interface using a direct graph metaphor where the user manipulates a small number of concrete objects to create his/her models;

• Use of a semi-quantitative language instead of mathematical (quantitative) linear equations to define relations among the elements (or objects) of a model;

• Automatic mapping of the model created by the user onto an internal mathematical model (always hidden from the user) which can be iterated over time, giving a visual feedback for the user;

• Wide range of modelling possibilities related to primary/secondary school (and undergraduate) curriculum with a high level of accuracy in modelling representation and output.

11.3.3 Format of the Studies

The format of the studies, particularly the Core study, can be used for similar investigations of other computer exploratory software.

The tasks proposed, together with the material given to the students and the analysis of the studies, can also be used as a first guide to introduce LinkIt in classroom activities (The format of the studies are presented in Sections 5.2 and

5.3 (Preliminary study) and Chapters 7 and 8 (Core study). The materials given to the students are in Appendices A and B).

11.4 FINDINGS

This section presents a synthesis of the main findings related to the Core Study (Chapters 7 to 10). It is divided into 3 subsections. The first two subsections are related to the two research (sub)questions presented in Figure 11.1. The last subsection is about the differences between the two age groups.

Before moving to the next sections, I present, as examples, two models the students worked with during the Core study. They illustrate the general nature of models students can construct themselves, or can use to generate discussion and thought.

Figure 11.2 shows a model about migration to big cities constructed by Group C as part of an expressive task presented to them in the very beginning of the second meeting. This model is quite typical of what the students did during this task and it includes feedback loops, different types of variables and links. In the case of Group C, after creating the model, they tried some simulations and made further changes in some parameters (also creating a new variable called *willing to move*) in order to make the model work closer to their ideas.

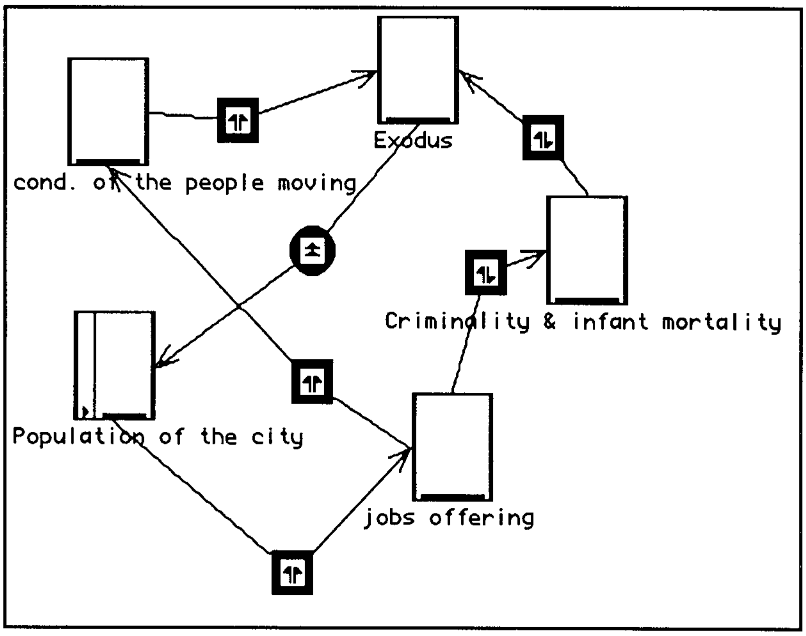


Figure 11.2: Model about migration to big cities created by Group C at the end of the first meeting

Figure 11.3 shows a model about how a refrigerator works. This model was presented to the students during an exploratory task (third day). This model presents many different features of the system working together: different ranges and types of variables; different strengths, direction and types of links; use of 'change by itself' (attached to the variable *sun*); feedback loops; etc. With this model on the screen, the students had to discuss its behaviour (the model as a whole and its parts), introduce some new variables and later construct a model about an air-conditioning. During the work with this model it was particularly remarkable how some students' ideas about controlling the temperature of a refrigerator evolved.

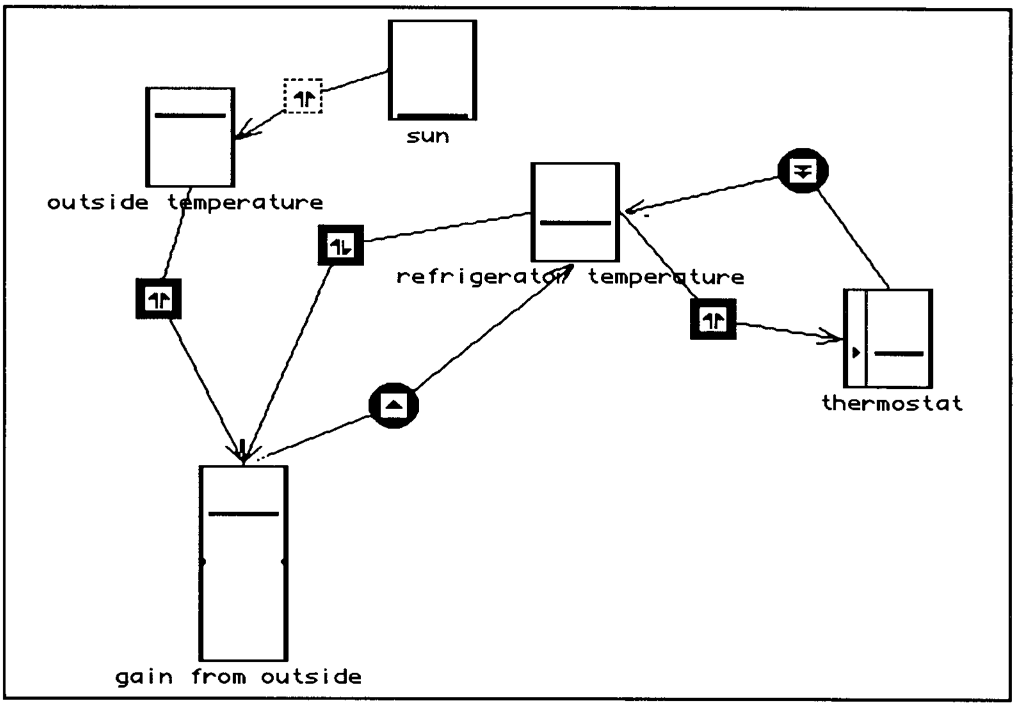


Figure 11.3: Model about a refrigerator working showed to the students during an exploratory task

11.4.1 Students Using the System

The general impression given by the analysis of students learning to use the software is that they understood, after a short period of training, the general ideas about how to manipulate the interface and the existing building blocks and their properties, being able to use them to construct and explore models.

A bird's eye view of what the students did when using LinkIt will give the impression that they did not learn everything about the system during the first meeting (although by the end of this meeting they were able to construct models by themselves). During the following meetings they made many steps forward and some backwards about understanding and manipulating the building blocks of the system to construct and explore models. This behaviour suggests that the software was neither very simple and trivial for the students, nor something out of their reach. What we observed there seem to be the youngsters being active learners, engaged in a process of constructing meaning for what they were doing at the same time as learning to use a new tool to do so.

However, if we zoom in to some particular aspects of the software it leaves the impression that the students (sometimes) did not perceive the system components in the same way as they were defined. Two instances of this behaviour were: (1) when the students simply misunderstood how a certain functionality of the system operates; and (2) when the students employed tangential aspects of the problem/system to decide about the use of a certain object/property of the system.

In the first case are the examples where some students did not notice the difference made by the system between the 'opposite' and 'inverse' direction relationships. Although the students' behaviour can be partially attributed to the way variables work in LinkIt, it is also certainly related to their daily life experience where these two ideas are mixed up. However, in this case, the use of the system in an educational setting can provide interesting opportunities for students to investigate the differences between these two mathematical relations.

An example of the second case (and which happened many times) is about how students chose between 'cumulative' and 'go together' relationships. Instead of using pure mathematical reasoning about the behaviour of variables and relationships, they normally justified them using arguments that were very much qualitative and strongly based on what they perceived from the visual feedback given by the system. This behaviour again reinforces a positive aspect of the system: bridging students' ideas to a more formalised way of seeing the world.

11.4.2 Thinking and Learning with the Software

An important aspect of LinkIt as an educational tool is that it presents learners with a set of building blocks that may allow them to externalise and test their internal ideas, helping them to shape their inner thoughts and become active builders of their own intellectual structures. Although the studies in this thesis were not designed to investigate whether learning new curriculum ideas occurred, they nevertheless provided evidence that some reasoning abilities required for the development of students' scientific skills arose (see Chapter 10). For example, there were situations where the students - when reflecting upon what was on the screen - were able to propose and implement some sophistication in their models, permitting them to arrive at a more complex level of articulation of their ideas.

Another important characteristic in the way the students used the system to think about the problems proposed to them was the existence of a pattern to

construct/discuss their models. As discussed in Section 10.2.3 the students tended to follow a process of successive refinement relying on the visual feedback of the system. This process basically consisted of three stages:
(1) Construction of a "laundry list" - The intention here is to produce a "draft" of their ideas, putting on the screen the relevant variables and their causal relations;
(2) Fixing the model - Start doing tests and simulations in order to make variables and links work according to their ideas;
(3) Sophistication of the model - Discuss and introduce new relevant ideas to the problem.

This characteristic pattern of dealing with the modelling process can have important consequences for those interested in the (re)design of the system as well as those concerned with its educational use. For the first group of people it raises questions about how to improve/facilitate the students' work in each stage (Does it make any difference to make the system to better suit each of the stages ? What kind of tools/functionality should be available at each stage ?). For the second group it raises questions about how to exploit this pattern of working with models to achieve a better understanding of systems dynamic and modelling process.

11.4.3 Differences Related to the Age Difference

Another important finding of the studies is that considerable differences were not found between the younger and older students when using the system to perform the tasks proposed. Bearing in mind that the younger students were about 13-14 and the older ones 16-17 this might seem surprising. It may perhaps be explained in several ways. First, it indicates that the system is sufficiently friendly and easy to use so as not to present serious difficulties even for younger pupils. Because it was new to both groups, and required a kind of thinking neither had experienced before, both groups had similar responses to using it. Secondly, the students worked with the system for a fairly short period of time. It seems probable that one could, with more extended use of the system, move into areas of subject matter (e.g. complex mechanical systems) where differences in understanding between the two age groups could be expected to appear. Further, in models requiring more extensive use of the system (e.g., more use of 'multiply' links) it could again be that differences would appear.

11.5 LIMITATIONS

The limitations of this work have to do with:

- The design of the software

• The format of the studies and its implications for the findings.
These topics are discussed separately in the following subsections.

11.5.1 Design of the Software

The software was designed around two main elements - variables and links - and a set of rules that control their behaviour. Although most adults do not have too much problem in thinking about the world in terms of variables and relationships, related literature about reasoning with modelling systems (including this work) and about commonsense reasoning suggests that young students do not see the world in this way. Therefore, in a certain way, the way the two main building blocks of the system work could represent a constraint on the way young students use the system to externalise their ideas about the "modelled world". If the intention is to develop software through which students' ideas could be made concrete and testable, other possibilities of representation and combination of these elements should be considered. An example of this case pointed out by the Core study, is the phenomenon of reversibility of the links where some students thought it should be possible to calculate the value of an independent variable from its dependent factors (see topic "Mutual constraint" in Section 9.2.2.1).

11.5.2 The Format of the Studies and its Implications for the Findings

The two exploratory studies carried out in this research involved a small number of groups working for a limited period of time. Therefore the findings found here can only be seen as possibilities that may happen with students when engaged in similar tasks.

The choice of the problems and the way the tasks were introduced to the students were supported by some assumptions about learning followed by the researcher. Basically it was assumed that the introduction of a tool to new users can be more effective if it is grounded on the previous knowledge of that user and following a "hands on" style of work[1]. This had direct consequences for the analysis of the data and its findings. In relation to the investigations about the design and educational values of LinkIt, it could be said that the first study was designed more to see whether the ideas existing in the software worked well than to offer a precise justification for using a style "X" of interface instead of an "Y" one. During its preparation it was expected that the analysis would raise more questions about the design of the software and would point to some modifications and addition of new features to the software, and indeed, this was

[1] This adopted style does not imply that it is the only way of introducing the software to students nor the only possibility of discussing science related subjects when a computer modelling tool is available.

what happened. The second version of the software was developed based on the findings of the Preliminary study. The Core study - which used LinkIt II - did not put an end to the guiding research (sub) question "How can someone's ideas be translated into computational terms ?". It suggested further new features that if added to the system could extend its modelling possibilities and (possibly) make it easier to use (see Chapters 9 and 10. Section 11.6.1 summarises the main features to be implemented).

11.6 FUTURE WORK

There are two main aspects related to this work that can be considered for future investigation: the design aspects of LinkIt; and preparing teachers for using the software. Each of these aspects are discussed in the sections that follow.

11.6.1 Design Aspects

The findings presented in the studies about how the students understood and used the system and its objects, suggest some important changes in the software that would improve its performance and students' work. A summarised list of these is presented below [2]:

Output of 'On/Off' variables

Some students had problems with the output of 'On/Off' variables and on some occasions suggested new possibilities for them. An interesting way to overcome these problems and make this type of variable more powerful would be to provide students with more choices for its output. Two of these possibilities are: (1) to create a fourth option for the output of 'On/Off' variables where no output is generated when the variable is "off" (this option could also automatically set to 'asleep' the outgoing links of the variable in question while it was "off"); (2) to change the default output when the variable is triggered to 'maximum' (instead of 'equal to ').

Representing variables on the screen

All variables in the system, no matter what entity of the "modelled world" they represent, have approximately the same appearance on the screen (a box with 1 or 2 levels). The use of different icons (or categories of icons) to represent different types of variables (e.g., a sun to represent a font of energy, a thermometer to represent temperatures) could be a possible way to shorten the

[2] Most of the points presented here are based on the observations made when the students were interacting with the system and are presented in Chapters 5, 9 and 10.

gap between students' concepts about the entities to be represented and the more abstract way the system uses to represent them.

Use of 'opposite direction' links

Although some fundamental changes about the representation of 'opposite direction' links were made, it still remained a source of problems for the students. A possible way to overcome this could be investing in a more careful way to introduce them to students, showing them more examples under different conditions and drawing their attention to their particulars, instead of simply looking for better interface representations.

Reversibility of the links

Some students envisaged the "reversibility of the links" where it would be possible to calculate the value of a causal variable from its dependent factors. Although the implementation of this idea could expand the modelling possibilities of the system it would certainly imply an increase in the complexity of the system (e.g. creating a new category of "both direction" links), which would compromise the trade off between the simplicity of the system and its modelling possibilities.

Graph option

Despite the importance of usage of graph representation of variables in science, LinkIt II does not support this facility. To incorporate it into the system, some design issues have to be considered. First is the fact that the number of iterations of a simulation with LinkIt is not pre-determined. The user is in total control of how far a simulation can go by saying in real time when it has to stop. However, in order to present a graph in a limited screen region (possibly a small window inside the working area) it is necessary to compute a scale, which implies that the number of iterations has to be fixed in advance. Another important fact is that the system squashes the resultant values of the variables within a certain interval in order to represent them within the frontiers of a variable box on the screen. However when it comes to the graph representation, a fundamental question is posed: Does the graph of a certain variable have to represent the behaviour of its level on the screen or the behaviour on the variable in relation to its inputs ?

Repeated simulations with the same initial values

During the experiments with LinkIt I and LinkIt II a need was found for a facility that could permit repeated simulations using the same initial values. This facility is particularly interesting to help students to follow the behaviour of some variables within their models containing a large number of variables (5 or more) or with feedback loops. This facility can be easily implemented and does not compromise the design of the system as it is.

Algorithm Used in the Simulation

LinkIt employs a variation of Euler's method to solve the ordinary differential equations that govern the behaviour of the variables during a simulation (see Sections 4.5.3.3 and 6.3.1). Although this method is non conservative, its choice was mainly based on the fact that it is easier to compute and faster than other methods like Runge-Kutta. Three years ago when the software was developed and when only slow machines were available for its development and use, these arguments were decisive for its choice. Nowadays with more powerful computers available in most of the schools, it makes sense to change the part of the inferential motor of the system that deals with the iteration of the model to use a method such as 4Th order Runge-Kutta[3].

Another important aspect that is beyond the results of the experiments done here is to think about integrating the software with other tools. Some possibilities are:

- Integrate the software with an hypertext environment - Within this new environment students would be able to be more explicit about the variables created and make predictions before testing. Those hyperdocuments created could be helpful for them to reflect on their own process of evolving ideas about the model as whole and its parts.
- Integrate different models - This facility could permit students to work in different sections of the same model and provide them a way to connect these different parts, running them separately or together.
- From semi-quantitative to more quantitative thinking - Ultimately the schools we have today give a high value to quantitative thinking and manipulation of analytical equations. So from this perspective LinkIt should bridge the student from one way of thinking to the other. At present I can see two alternatives here. The first one is to start using more quantitative modelling systems that already exist in the market (e.g., STELLA) as the students gain more confidence about modelling and develop their mathematical skills. This way of thinking is more related to the pedagogical aspects of using computer modelling in education than to the redesign of the software. A second possibility, which remains an open question, is whether there could be a computer modelling tool that supports both semi-quantitative and quantitative ways of thinking.

---

[3] 4th order Range-Kutta - is a conservative method. However in comparison to Euler's method it requires four additional evaluations at each iteration. Depending on the speed of the computer being used this can slow down the visual feedback of the dynamic behaviour of the variables on the screen, compromising the dynamics of the system as whole.

11.6.2 <u>Preparing Teachers for Using the Software</u>

A important aspect that was out of the scope of this work, is about how to introduce LinkIt in an educational setting. It is important to note that LinkIt is an educational tool that carries no specific content to be taught and as such it is inappropriate to be highly prescriptive about how it should be used, risking limiting the possibilities of the system. However, of prime importance is to make teachers aware of the full power of the software. Some of the most relevant aspects that need to be considered for their preparation are discussed below.

(1) Related to teacher training - Ultimately the time available to train teachers will dictate how to train them. Also the level of involvement of the school and teacher interests will be of great importance for the future use of the software in the classrooms of a particular school.

- How are teachers going to be trained (INSET courses, postgraduate courses, etc.) ?
- What is the level of interest of the schools from where the teachers come ?

(2) Related to the pedagogical uses of the software - LinkIt can be used for different goals ranging from a simple demonstration of a simulation of a particular phenomenon to interdisciplinary project based work. The teachers should be clear about the pedagogical possibilities of the software (and how these possibilities can be implemented) in order to better decide about how and when to use it.

- What are the pedagogical aims of using the software in classrooms ?
- Different results can be achieved with different uses of the system (e.g., LinkIt can be used as an alternative way to the blackboard to demonstrate the dynamic behaviour of a system, to develop project-based programmes; etc.). What are the most likely uses of the system in a particular educational set ?

(3) Related to school resources - The computer resources/time available per student impose a certain level of constraint on the different possibilities of use of the software.

- What are the computer resources in a particular educational setting and how do they constrain the usage of the software ?

These questions are not completely new insofar as other computer modelling systems are already been used in classrooms activities. Thus, a careful reading of other projects and research programmes involving computer modelling in education is of great importance. For those interested in the subject I would suggest a review of the STACIN project (Mandinach & Cline, 1994), the work with ScienceModeller (Blumenfield, Soloway, Marx, Krajcik, Guzdial, & Palincsar, 1991; Jackson, Stratford, Krajcik, & Soloway, 1995) and the present work of Forrester at MIT (www site: http://sysdyn.mit.edu/).

11.7 FINAL REMARKS

The idea of looking for evidence to demonstrate the importance of LinkIt as an educational tool has been with me since the beginning of the design of its first version. Even after the analysis of the Preliminary study I was still concerned about its educational use. At that time, I was aiming to design the Core study in such a way that it would be possible to prove that the software had a strong educational value and that, indeed, students could engage in a deep process of learning while using it ! However, during the time I was redesigning LinkIt and proceeding with a further literature review, I became aware that the idea of "engaging in a deep process of learning", although frequently used in the writings about education, is very difficult to demonstrate in practice. To demonstrate something close to that idea would require some months of working with LinkIt. However, what was possible, considering the time constraints of the research, was to gather some evidence about the educational potential of the software.

The educational value of LinkIt became evident to me much earlier than the end of the analysis of the Core study. It happened when I was still in Brazil collecting the data. After the end of the last interview with one group, I kept the tape recorder on and started chatting with the students about their impressions of using the software in "real classroom activities". This is what they said:

> (talking about the importance of the software)
>
> D: " *Because no other system exists that can do this* (that can do what LinkIt does)"
>
> R: " *It portrays the reality...*"
>
> D: " *It portrays what we want to be portrayed... if we want it.*"
>
> ...
>
> R: " (it can be used to) *represent* (something)*better. Sometimes the teacher just explains and that is all...When we can't understand something, we can understand* (it by) *seeing the screen of the computer*"

For the students, the system has educational value because it helps them to represent ("portrays") and understand ideas.

BIBLIOGRAPHY

Apple Computer. (1980). Apple SuperPILOT Reference Manual. Apple Computer Inc.

Bliss, J., Mellar, H., & Ogborn, J. (1992a). Tools for Exploratory Learning Programme - Technical Report 3: Semi-quantitative Reasoning - Exploratory (ESRC Information Technology In Education Initiative - End of Award Report). King's College London, Imperial College, Institute of Education - London.

Bliss, J., Mellar, H., Ogborn, J., & Nash, C. (1992b). Tools for Exploratory Learning Programme - Technical Report 2: Semi-quantitative Reasoning - Expressive (ESRC Information Technology In Education Initiative - End of Award Report). King's College London, Imperial College, Institute of Education - London.

Bliss, J., & Ogborn, J. (1989). Tools for Exploratory Learning, A Research Programme. Journal of Computer Assisted Learning, 5, 37-50.

Bliss, J., Ogborn, J., Boohan, R., Briggs, J., Brosnan, T., Brough, D., Mellar, H., Miller, R., Nash, C., Rodgers, C., & Sakonidis, B. (1992c). Reasoning Supported By Computational Tools. Computers Education, 18(1-3), 1-9.

Blumenfield, P. C., Soloway, E., Marx, R. W., Krajcik, J. S., Guzdial, M., & Palincsar, A. (1991). Motivating Project-Based Learning: Sustaining the Doing, Supporting the Learning. Educational Psychologist, 26(3 & 4), 369-398.

Booth, P. (1990). An Introduction to Human-Computer Interaction. Hillsdale, NJ: Lawrence Erlbaum Associates.

Briggs, J., Nichol, J., Brough, D., & Dean, J. (1989). Linx88. London: PEG.

Brown, J. S., Collins, A., & Duguid, P. (1989). Situated cognition and the culture of learning. Educational Research, 18, 32-42.

Chen, D. (1994). An Epistemic Analysis of the Interaction between Knowledge, Education and Technology. In E. Barrett (Ed.), Sociomedia: Multimedia, Hypermedia and the Social Construction of Knowledge Cambridge, MA: MIT Press.

Clark, D. (1993). Self-Explanatory Objects: An Investigation of Object-Based Help. Unpublished PhD. Thesis. Open University.

Craik, K. J. W. (1943). The nature of Explanation. Cambridge: Cambridge University Press.

Crandall, R. (1984). Pascal Applications for the Science. New York: John Wiley & Sons.

de Corte, E. (1990). Learning with new information technologies in schools: perspectives from the psychology of learning and instruction. Journal of Computer Assisted Learning(6), 69-87.

de Kleer, J., & Brown, J. S. (1981). Mental models of physical mechanisms and their acquisition. In J. R. Anderson (Ed.), Cognitive skills and their acquisition (pp. 285-309). Hillsdale, NJ: Lawrence Erlbaum.

de Kleer, J., & Brown, J. S. (1983). Assumptions and Ambiguities in Mechanistic Mental Models. In D. Gentner & A. L. Stevens (Eds.), Mental Models. Hillsdale, NJ: Lawrence Erlbaum.

de Kleer, J., & Brown, J. S. (1985). A Qualitative Physics Based on Confluences. In J. R. Hobbs & R. C. Moore (Eds.), Formal Theories of the Commonsense World (pp. 109-183). Norwood, NJ: Ablex Publishing Corporation.

Dillon, C. (1994). Qualitative Reasoning about Physical Systems - An Overview. Studies in Science Education, 23, 39-57.

DiSessa, A. (1987). Artificial worlds and real experience. In R. W. Lawler & M. Yazdani (Eds.), Artificial Intelligence and Education (pp. 55-77). Norwood, NJ: Ablex Publishing Corporation.

Doulcet, P., & Sloep, P. B. (1992). Mathematical Modelling in the Life Science.

Draper, S., Driver, R., Hartley, R., Hennessy, S., Mallen, C., Mohamed, R., O'Malley, C. E., O'Shea, T., Scanlon, E., & Twigger, D. (1991). Design Considerations in a Project on Conceptual Change in Science. Computers and Education, 17(1), 37-40.

Driver, R. (1983). The Pupil as Scientists ? Milton Keynes: The Open University Press.

Driver, R. (1989). Changing Conceptions. In Adey, P. (Ed.), Adolescent Development and School Science (pp. 80-102). Falmer Press.

Erickson, T. D. (1990). Working with Interface Metaphors. In B. Laurel (Ed.), The Art of Human-Computer Interface Design (pp. 65-74). Addison-Wesley Publishing Company.

Forbus, K. (1984). Qualitative Process Theory. Artificial Intelligence, 24(1-3), 85-168.

Forrester, J. M. (1968). Principles of Systems. Cambridge, MA: Wright-Allen Press Inc.

Gentner, D., & Stevens, A. (Eds.). (1983). Mental Models. Hillsdale, NJ: Lawrence Erlbaum Associates.

Gilbert, J. K., & Osborne, R. J. (1980). The Use of Models in Science Teaching. School Science Review(62), 57-67.

Gomoll, K. (1990). Some Techniques for Observing Users. In B. Laurel (Ed.), The Art of Human-Computer Interface (pp. 85-90). MA: Addison-Wesley Publishing Company.

Gorny, P. (1988). Didatic and Software-Ergonomic Aspects of Dynamic Modelling and Simulation Systems. Computers in Education, 291-297.

Gutierrez, R., & Ogborn, J. (1992). A causal framework for analysing alternative conceptions. International Journal of Science Education, 14(2), 201-220.

Hartley, J. R., Byard, M. J., & Mallen, C. L. (1991). Qualitative Modelling and Conceptual Change in Science Students. In International Conference on the Learning Science, (pp. 222-230). Chicago: Northwestern University .

Hartley, R. J. (1981). MODL - an interactive computer system for mathematical modelling by non-specialists. Unpublished PhD. Thesis, Chelsea College, London.

Hebenstreit, J. (1987). Simulation as an Educational Tool: An Encounter of the Third Kind. Informatics and Education, v.75-v.82.

Hobbs, J. R., & Moore, R. C. (Eds.). (1985). Formal Theories of the Commonsense World. Norwood, NJ: Ablex Publishing Corporation.

Hoyles, C., & Sutherland, R. (1992). LOGO mathematics in the classroom (revised ed.). London: Routledge.

Jackson, S. L., Hu, J. T., & Soloway, E. (1994). ScienceWorks Modeler: Scaffolding the Doing of Science. In CHI'94 - Human Factors in Computing Systems, 1 (pp. 249-250). Boston: ACM press.

Jackson, S. L., Stratford, S. J., Krajcik, J., & Soloway, E. (1995). Learner-Centered Software Design to Support Students Building Models. In American Educational Research Association Meeting. San Francisco: USA

Johnson-Laird, P. N. (1983). Mental Models: Towards a cognitive science of language, inference and consciousness. Cambridge, MA: Harvard University Press.

Kozma, R. B. (1987). The Implications of Cognitive Psychology for Computer-Based Learning Tools. Educational Technology (November), 20-25.

Kuipers, B. (1994). Qualitative Reasoning: Modelling and Simulation with Incomplete Knowledge. Cambridge: The MIT Press.

Kurtz dos Santos, A. C. (1992). Computational Modelling in Science Education: A Study of Student's Ability to Manage Some Different Approaches to Modelling. Unpublished PhD. Thesis, University of London.

Kurtz dos Santos, A. C., & Ogborn, J. (1994). Sixth Form Students' ability to engage in computational modelling. Journal of Computer Assisted Learning(10), 182-200.

Kurtz dos Santos, A. C. (1995). Introdução a Modelagem Computacional na Educação. Rio Grande: Editora da Furg - Brazil.

Lakoff, G., & Johnson, M. (1980). Metaphors We Live By. Chicago: Chicago University Press.

Lindfield, G. R. (1992). The role of the STELLA software package in simulation and modelling. International Journal of Mathematical Education Science Technology, 23(6), 865-880.

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, N.J: Lawrence Erlbaum Associates.

Marx, G. (1984). Simulation games in science education. European Journal of Science Education, 6(1), 31-45.

Mellar, H. (1990). Creating alternative realities: computers, modelling and curriculum change. In P. Dowling & R. Noss (Eds.), Mathematics versus the National Curriculum (pp. 176-191). Basingstoke: Falmer Press.

Mellar, H., Bliss, J., Boohan, R., Ogborn, J., & Tompsett, C. (Eds.). (1994). Learning with Artificial Worlds: Computer Based Modelling in the Curriculum (1st. ed.). London: The Falmer Press.

Millar, R., & Driver, R. (1987). Beyond Process. Studies in Science Education, 14, 33-62.

Miller, R., Briggs, J., Brough, D., & Ogborn, J. (1992). Tools for Exploratory Learning Programme - Technical Report 1: Tools (ESRC Information Technology In Education Initiative - End of Award Report). King's College London, Imperial College, Institute of Education - London.

Miller, R., & Ogborn, J. (1993). Educational Tools for Computational Modelling. Computers and Education, 21(3), 205-261.

Miller, R. S., Ogborn, J., Turner, J., Briggs, J. H., & Brough, D. R. (1990). Towards a Tool to Support Semi-Quantitative Modelling. In International Conference on Advanced Research on Computers in Education Tokio, Japan.

Millwood, R., & Stevens, M. (1990). What is the Modelling Curriculum ? Computers and Education, 15(1-3), 249-254.

Nickerson, R. S. (1988). Technology in Education: Possible Influences on Context, Purposes, Context and Methods. Hillsdale, N.J.: Lawrence Erlbaum Associates.

Nicol, A. (1990). Interfaces for Learning: What do good Teachers know that we don't ? In B. Laurel (Ed.), The Art of Human-Computer Interface Design (pp. 113-122). MA: Addison-Wesley Publishing Co.

Norman, D. A. (1983). Some Observations on Mental Models. In D. Gentner & A. L. Stevens (Eds.), Mental Models (pp. 7-14). Hillsdale, NJ: Lawrence Erlbaum Associates.

Norman, D. A., & Draper, S. (Eds.). (1986). User-centered system design: New perspectives on human-computer interaction. Hillsdale, NJ: Lawrence Erlbaum Associates.

Novak, J. D., & Gowin, D. B. (1984). Learning How to Learn. Cambridge University Press.

Ogborn, J. (1990). A Future for Modelling in Science Education. Journal of Computer Assisted Learning, 6, 103-112.

Ogborn, J. (1992). Modelling With the Computer at All Ages. Portugal Physics, 21(3/4), 99-114.

Ogborn, J. (1993). Models and their Makers.

Ogborn, J., & Wong, D. (1984). A Microcomputer Dynamical Modelling System. Physics Education, 19, 138-142.

Papert, S. (1980). Mindstorms, children, computers and powerful ideas. NY: Basic Books.

Piaget, J. (1954). The construction of Reality in the Child. NY: Ballentine Books.

Resnick, L. B. (1983). Mathematics and Science Learning: A new conception. Science Education (april).

Richmond, B. (1987). An Academic User's Guide to STELLA. High Performance Systems, Inc. Lyme.

Riley, D. (1990). Learning about Systems by Making Models. Computers and Education, 15(1), 255-263.

Roberts, N. (1983). Introduction to Computer Simulation - A systemic dynamics modelling approach (1st. ed.).

Rubinstein, R., & Herssh, H. (1984). The human factor: Designing computer systems for people. Burlington, MA: Digital Press.

Rumelhart, D. E., & McLelland, J. I. (1987). Parallel Distributed Processing - Explorations in the Microstructure of Cognition. Volume 1: Foundations (3rd. ed.). Cambridge, MA: MIT press.

Rutkowski, C. (1982, An Introduction to the Human Applications Standard Computer Interface, Part I: Theory and Principles. Byte, 291-310.

Sampaio, F. F. (1995). Design, Development and Pilot Testing of a Modified Version of a Semi-quantitative Modelling Tool.Unpublished Upgrading Report, Institute of Education - University of London.

Sasse, A., Life, A., Cunningham, J., & Pitt, J. (1985). UCL/IC Human-Computer Interaction Joint Course. University College London.

Schecker, H. (1993). Learning physics by making models. Physics Education, 28(2), 103-106.

Shneiderman, B. (1992). Designing the user interface: strategies for human-computer interaction (2nd. ed.). MA: Addison-Wesley Publishing Company.

Shute, V. J., & Gawlick-Grendell, L. A. (1994). What Does the Computer Contribute to Learning ? Computers and Education, 23(3), 177-186.

Smith, D. C., Irby, C., Kimball, R., & Verplank, B. (1982) Designing the Star User Interface. Byte, 242-282.

Solomon, C. (1987). Computer Environments for Children - A Reflection on Theories of Learning and Education. MA: The MIT Press.

Sowa, J. F. (1984). Conceptual Structures Information Processing in Mind and Machine. MA: Addison-Wesley Publishing Company.

Staudenmaier, H-M. (1982). Use of computers in science education. European Journal of Physics, 3, 144-151.

Steed, M. (1992). STELLA, A Simulation Construction Kit: Cognitive Process and Educational Implications. Journal of Computers in Mathematics and Science Teaching, 11, 39-52.

Teodoro, V. D. (1992a). GALILEO - Direct Manipulation of Physical Concepts in a Computerized Exploratory Laboratory Universidade Nova de LIsboa: Lisbon.

Teodoro, V. D. (1992b). A Model to Design Computer Exploratory Software for Science and Mathematics. In NATO-ARW: The Use of Computers Models for Explication, Analysis and Experiential Learning. Chateau de Bonas at Castera-Verduzan/Gers: France.

Teodoro, V. D. (1994). Learning with Computer-Based Exploratory Environments in Science and Mathematics. In S. Vosniadou, E. de Corte, & H. Mandl (Eds.), Technology-Based Environments (pp. 26-32).

Tognazzini, B. (1990). Consistency. In B. Laurel (Ed.), The Art of Human-Computer Interface (pp. 75-77). MA: Addison-Wesley Publishing Company.

Toval, A., & Flores, M. (1987). Computer Systems Simulation in Education: Description of an Experience. Computers and Education, 11(4), 293-303.

Vitale, B. (1988). Psycho-Cognitive Aspects of Dynamic Model-Building in LOGO: A Simple Population Evolution and Predator/Prey Program. Journal of Educational Computing Research, 4(3), 227-251.

Vygotsky, L. (1962). Thought and Language. Cambridge, MA: The MIT Press.

Webb, M. (1990). Learning by Building Computer Based Qualitative Models. Computer Education (66), 6-8.

Webb, M., & Hassell, D. (1988). Opportunities for Computer Based Modelling and Simulation in Secondary Education. Computers in Education, 271-277.

Whitfield, A. H. (1988). STELLA and its Impact on the Teaching of Mathematical Modelling. Computers in Education, 299-304.

Wild, M. (1996). Mental models and computer modelling. Journal of Computer Assisted Learning, 12, 10-21.

Wong, D. (1986). Development and Evaluation of a Microcomputer Dynamic Modelling System for Teaching A - Level physics. Unpublished PhD. Thesis, University of London.

Clancey, W. J. (1992). Representation of Knowing: in defense of cognitive aprenticeship. Journal of Artificial Intelligence in Education, 3, 139-168.

O'Shea, T., & Self, J. (1987). Learning and Teaching with Computers. Sussex: The Harvest Press.

Wong, E. D. (1993). Self-generated Analogies as a Tool for Constructing and Evaluating Explanations of Scientific Phenomena. Journal of Research in Science Teaching, 30(4), 367-380.

APPENDIX A

PRELIMINARY STUDY: SUPPORT TEXTS GIVEN TO THE STUDENTS

Two support texts were given to the students during the Preliminary study. The first text was about the biosphere and water-cycle (used in Task 2) and the second text was about the nomad-population (Task 3).

TEXTO SOBRE BIOSFEREA

A biosfera é uma estreita camada envolvendo o planeta Terra onde todas as formas de vida existem. Os principais elementos que compõem a bioesfera são o ar, a água e o solo .

A biosfera é composta de diversos sistemas auto-reguláveis que controlam o funcionamento do nosso planeta.

No modelo a seguir, vamos olhar um destes sistemas auto-reguláveis, responsável por manter em equilíbrio a quantidade de água existente no planeta e a sua temperatura.

UMA EXPLICAÇÃO PARA O SISTEMA AUTO-REGULáVEL DE AGUA EXISTENTE NO PLANETA E SUA TEMPERATURA

Quando o sol brilha sobre uma superfície de água (oceanos, lagos) , o calor emitido causa a evaporação de parte da água existente . O vapor sobe para o céu , formando as nuvens. Os ventos empurram as nuvens para diferentes partes do Planeta e em alguns casos estas nuvens se transformam em chuvas. Desta forma, a água evaporada retorna a superfície da Terra caindo sobre os oceanos, lagos e solo. Assim que as nuvens se dissipam, o sol volta a brilhar sobre as superfícies de água e a evaporação recomeça. Este processo é também responsável pelo controle de temperatura do planeta Terra.

Quando o sol, através do seu brilho, esquenta a Terra acontecem, como já vimos, a evaporação e formação de nuvens no céu. As nuvens, por sua vez, impedem que o brilho do sol chegue à superfície da Terra, fazendo com que a temperatura na superfície diminua. Quando a temperatura diminui, as nuvens se transformam em chuvas e somem, permitindo denovo que o sol brilhe sobre a superfície do planeta e o ciclo recomece.

Este sistema auto-regulável garante que todos os seres vivos terão sempre água e uma faixa de temperatura constante para a sua existência.

(Texto baseado no livro de Ciências adotado pela Escola)

Support text 1: The text addresses the problem of biosphere and gives a brief account about the water-cycle

CARREGUE O MODELO BIOESFERA E VERIFIQUE SE ELE CORRESPONDE ÀS IDÉIAS APRESENTADAS ANTERIORMENTE E ÀS SUAS IDÉIAS DE COMO FUNCIONA A NOSSA BIOSESFERA .

Worksheet 1: Ask the student to load the model "BIOSFERA" and to compare it with their own ideas and what is written in the support text 1

TEXTO SOBRE POPULAÇAO NOMADE

Uma grande tragédia aconteceu numa área do continente Africano abaixo do deserto do Saara, conhecida como Sahel. Esta região tem sido habitada há vários séculos por populações nomades.

Devido `as suas condições climáticas, o Sahel nunca foi uma região fácil de viver. No entanto, os nômades têm conseguido sobreviver nesta área de uma forma impressionante.

A cada nova estação eles movem os seus rebanhos para novos campos de pastagem. Devido `as limitações de comida e água os rebanhos nunca crescem muito. Estas mesmas limitações, aliadas `as doenças e ao clima hostil, fazem também com que a população nômade não cresca muito. Mais ainda, a cada 30 anos aproximadamente uma forte seca acontece, causando a morte de um grande número de animais e pessoas, evitando sempre que as populações de animais e pessoas cresçam muito.

Há alguns anos, pessoas trabalhando para organizações mundiais como a ONU decidiram tentar melhorar a qualidade de vida destas populações nômades. Duas principais atitudes foram tomadas. Primeiro, foram introduzidas técnicas modernas de medicina tais como vacina contra a malária e doença do sono. Algumas doenças animais também foram controladas. Segundo, através da utilização de novas tecnologias para perfuração de poços profundos, uma farta quantidade de água ficou disponível. Este aumento do suprimento de água, possibilitou o aumento do rebanho. No entanto, este aumento repentino do rebanho fez com que a comida disponível para os animais (campos de pastagem) logo se extinguisse, fazendo com que os rebanhos começassem a morrer de fome. Com a morte dos animais, muitos nômades ficaram sem alimentação, terminando também por morrerem de fome.

A ONU novamente estava diante de um severo problema, desta vez bem mais grave que o original.

(Texto baseado no original apresentado em Roberts (1983)

Support text 2: The text addresses the problem of a nomad population living in the Sahel desert and the interference of the U.N. on their way of living.

CONSTRUA UM MODELO QUE REPRESENTE AS IDÉIAS APRESENTADAS ACIMA.

SUGESTÃO: Procure assinalar no texto os principais itens que contribuem para a existência e extermínio da população nomade e seus rebanhos.

Worksheet 2: Ask the students to construct a model to represent the ideas presented in the support text 2. It also gives a suggestion to start thinking about the main elements that contribute to existence and extermination of the population and its herds.

APPENDIX B

CORE STUDY:
WORKSHEETS AND SUPPORT TEXTS
GIVEN TO THE STUDENTS

Um dos grandes problemas encontrados nas grandes cidades é o problema de poluição em geral: do ar, dos rios e mares, sonora, visual, etc.

Focando especificamente o problema da poluição do ar, gostaria que voces pensassem sobre o problema e respondessem as seguintes perguntas abaixo:

1) NA SUA OPINIÃO, quais são as principais CAUSAS e CONSEQUENCIAS da poluição ?

2) Gostaria que você descrevesse como você acha que funciona o fenômeno da poluição do ar em grandes cidades ?

3) Como VOCE acha que esse problema pode ser resolvido (ou amenizado) ?

4) NA SUA OPINIÃO, de que forma a população da cidade pode atuar para tentar resolver este problema ?

Worksheet 1: The text addresses the problem of pollution in big cities but does not say anything about its causes and consequences.

The questions made are:

1) In your opinion what are the main causes and consequences of pollution of the air in big cities ?

2) I would like you to describe how do you think the phenomena of pollution in big cities works

3) How do you think this problem can be solved (or diminished) ?

4) In your opinion how the population of the city can help to try to solve this problem ?

Um problema enfrentado pelas grandes cidades é o de migração de populações para estes grandes centros. Quando essa população migrante vem essencialmente do campo, este fenômeno é conhecido como êxodo rural.
O problema de migração para grandes centros não é algo do nosso século. A época da "Revolução Industrial" (séc. XVIII) é um exemplo do passado onde grandes centros europues tiveram que encarar este problema.
O fenômeno da migração inicia-se normalmente pela atração exercida pelas grandes cidades através da oferta de empregos, melhores salários ou melhores condições de vida. Quando a população da cidade cresce muito e esgota a oferta de emprego ou faz com que ela desça a n'veis muito baixos, temos então que uma boa parte desta população fica sem trabalho, causando sérios problemas para os mesmos e para a cidade em geral, aumentando, entre outros, os indices de falta de moradia, criminalidade e mortalidade infantil.

Support text 1: This text talks about the problem of migration to big cities. It mentions some attractive factor like jobs, salaries and life conditions. It also mention some problems (consequences): Homeless, Crimes, Infant mortality

Uma caracteristica importante encontrada na Natureza é a do equilibrio existente entre as espécies.
Um exemplo t'pico é o da relação entre coelhos e raposas numa floresta onde não existe interferência humana.
É sabido que nestes locais os coelhos são a fonte principal de alimento das raposas. Assim, quando as raposas comem os coelhos, elas podem crescer e procriar. Mas quando a população de raposas começa a crescer, mais e mais coelhos são necessários para alimentá-las, chegando a um ponto em que já não existem mais coelhos suficientes para alimentar todas as raposas. Elas então tem que competir pelos poucos coelhos existentes e aquelas raposas que não conseguem se alimentar direito acabam morrendo, diminuindo sensivelmente a população de raposas. Com a diminuição da população de raposas, os poucos coelhos restantes passam a ter mais chances de procriar e aumentar a sua população existente.

(Baseado no livro Mathematical Modeling in the Life Science Doulcet, P., Sloep, P. B., (1992))

Support text 2: This text describes the problem of predator-prey using the example of foxes and rabbits. It does not make a clear account of the variables presented in the model

Perguntas:

1) Existe alguma situação em que os coelhos acabam ?

2) Existe alguma situação em que as raposas acabam ?

3) Voce acha que o modelo esta' representando corretamente o problema ? Explique sua resposta

4) Voce conseguiria desenhar um grafico da populacao de coelhos em relação ao tempo baseado no seu conhecimento do problema ?

5) Voce conseguiria desenhar um grafico da populacao de raposas em relacao ao tempo baseado no seu conhecimento do problema ?

6) Voce acha que o modelo deve ser mudado para que fique de acordo com as suas idéias sobre o problema ? Se SIM, quais as mudanças que voces querem fazer ? Se NAO, explique porque voce acha que o modelo não precisa ser alterado

Worksheet 2: Questions about the Predator-Prey problem: 1) Is there a situation where the rabbits finish ? 2) Is there a situation where the foxes finish ? 3) Do you think the model is representing the problem correctly ? Explain your answer 4) Could you try to construct a graph about the rabbits population against time based on the model and your knowledge about the problem ? 5) Could you try to construct a graph about the foxes population against time based on the model and your knowledge about the problem ? 6) Do you think that the model has to be changed in order to be in accordance with your ideas about the problem ? If YES, what are the changes you want to do ? If NO, explain why do you think the model doesn't need to be changed

Os seres humanos e muitos outros vertebrados possuem um complexo e delicado sistema visual composto de vários mecanismos. Um desses mecanismos, conhecido como reflexo da pupila, é responsável por regular a quantidade de luz que entra no olho através da mudança de diâmetro de abertura da pupila. A seguir é apresentado uma descrição mais detalhada deste mecanismo.
Os objetos que nos rodeiam refletem a luz proveniente do sol ou de uma fonte artificial como uma lâmpada. Estes raios de luz têm uma certa intensidade, podendo variar entre muito forte (ex. luz direta do sol num dia de verão sem nuvens) e fraco (a luz de uma vela posicionada a 100 m de distância). De acordo com a intensidade destes raios de luz, a pupila tenta se acomodar, aumentando ou diminuindo a sua área, de tal forma a permtir que s— penetre no olho uma quantidade de luz que não prejudique o sistema de visão. Este sistema é capaz de operar de forma bem rápida sendo que o seu n'vel de conforto é diferente para cada uma das espécies. Os gatos por exemplo, têm um nivel de conforto maior que os seres humanos.

(Baseado no livro Mathematical Modeling in the Life Science Doulcet, P., Sloep, P. B., (1992))

Support text 3: Explains, in a qualitative way, how the eye-pupil reflex system works. All variables that belong to the model are mentioned here

1a) O que acontece com a pupila do olho humano quando uma pessoa está na praia num dia de sol ?
1b)Como voce testaria isso usando o modelo ? (Faça os testes com o modelo antes de responder)
1c) O que acontece com a pupila do olho humano quando uma pessoa está na praia num dia de sol e coloca um —culos escuro ?
1d) Como voce testararia isso usando o modelo ? (Faça os testes com o modelo antes de responder)
2) Como voce mostraria no modelo o que acontece com o olho humano quando uma pessoa sai de um tunel longo (tipo Rebouças) num dia de sol ?
3) Voce pode descrever o que representa e como funciona cada uma das caixinhas existentes no modelo ?

Worksheet 3: Questions about the eye-pupil system: 1a) What happen to the eye-pupil when someone is on the beach ? 1b) How do you show these using the model ? 1c) What happen to the eye-pupil when someone is on the beach and put a sun-glasses ? 1d) How do you show in the model the act of someone putting a sun glasses ? 2) How do you show in the model what happen to the eye when someone get out of a long tunnel in a sunny day ? 3) Can you give a description of what represents and how each box of the model works ?

Perguntas:

1a) O que voces prevem que vai acontecer com geladeira num dia de muito sol ?

1b) Como voces podem testar essa sua previsao no modelo ?

Worksheet 4: Questions about the Refrigerator problem: 1a) What do you predict is going to happen to the refrigerator in a very sunny day ? 1b) How can you test your prediction using the model ?

O que é uma alimentação sadia ? A maioria de nos não sabe. Isso não é porque somos estúpidos ou ignorantes ou simplesmente porque não nos importamos. É porque nos estamos confusos. Existem tantas informações conflitantes sobre alimentação sadia e saúde que não podemos reclamar quando "jogamos a toalha" e comemos uma pizza. Mas na verdade, no meio desta montanha de propagandas anunciadas pelas indústrias de alimentação, existem informações mais precisas sobre o assunto. O conselho outrora dado pelos nossos avós - três refeições ao dia com iguais quantidades de alimentos dos quatro principais grupos de alimentícios (carbohidratos, proteínas, gorduras e fibras) - tem-se mostrado inadequado. Apos anos de pesquisas na área de nutrição, novas recomendações oficiais tem sido anunciadas como uma boa dieta alimentar. Estas recomendações tem a forma de uma "piramide alimentar". A maior parte da pirâmide - a base - deve ser constituída de grãos integrais, que devem constituir 30 a 45 % da sua alimentação. A próxima camada deve ser constituída por vegetais que devem ser responsáveis por 15 a 20 % de sua alimentação, seguido pelas frutas - 10 a 15 % - , carnes e laticínios - 10 a 15 % - e por fim gorduras, óleos e doces que não devem perfazer mais do que 5 % do total de alimentos ingeridos.

A mensagem dada é clara: alimentação constituída de poucas gorduras e muitas fibras. Dizem os especialistas que a "receita" é boa tanto para perder peso quanto para ter uma vida mais longa e sadia.

Mas será que só os cuidados com a alimentação são suficientes ? E a vida sedentária, o stress e a poluição e outros fatores a que estamos submetidos, também não contam ?

(Baseado no livro The good nutrition guide - Options Magazine, (1995))

Support text 4: Talks about the problem of what is a good diet. In the beginning the text says that nobody knows well about this problem but later tells what is considered a good diet today (mentioning percentages) and its consequences. But in the last paragraph pose a question whether it is enough to have a good life condition, mentioning some other factors like stress, pollution, etc.

Um dos grandes problemas encontrados nas grandes cidades é o problema de poluição em geral: do ar, dos rios e mares, sonora, visual, etc.
Focando especificamente o problema da poluição do ar, gostaria que voces pensassem sobre o problema e respondessem as seguintes perguntas abaixo:

1) Gostaria que voces criassem um modelo de como voces acham que funciona o problema da poluição do ar em grandes cidades

2) Como VOCES acham que esse problema pode ser resolvido (ou amenizado) ?

3) NA SUA OPINIÃO, de que forma a população da cidade pode atuar para tentar resolver este problema ?

Support text 5: The text address the problem of pollution in big cities but does not say anything about its causes and consequences. The questions made are: 1) Could you create a model about your ideas on how the problem of pollution of the air in big cities happens; 2) How do you think this problem can be solved (or diminished) ?; 3) In your opinion how the population of the city can help to try to solve this problem ?

APPENDIX C

PRELIMINARY STUDY: SAMPLE OF A CASE STUDY

INTRODUCTION

This Appendix presents a sample of a case study related to the Preliminary study. The group presented here (Group D) is composed of a girl, Renata (17) and a boy, Gabriel (16).

From my point of view this group was one of the most participative in the study and presented good examples of "scientific thinking".

It is worth saying that we cannot observe these characteristics just by looking at the models they constructed, mainly because their models, in terms of complexity and correctness, were essentially the same as the ones made by the other groups. So, during the analysis of this group I will also try to deal with their response to the system.

When this case study was written in the beginning of 1995, I organized the text around relevant topics for the research instead of following a chronological description of what happened when they were interacting with LinkIt I. In this sample here I decided to keep this way of writing in order to give to the reader a better feeling of how the results of the Preliminary Study were achieved.

LEVEL OF ABSTRACTION

Like other groups (e.g. Group C) when we began to talk about pollution in order to create a model they tried to consider different aspects of the events involved in the process. Before the passage below we had agreed about two causes of pollution: *chemicals* (from the industries) and *smoke of the cars.*

> F: " (...) *and how do they cause pollution ? I mean, more chemicals bigger.... The more chemicals produced or more smoke produced bigger the pollution, or it is less, less smoke* (you have), *the more pollution. How is that relation of cause and effect ?*"
> G: " *It depends on* (the kind of) *smoke*"
> F: " *Depends on the smoke ?*"
> R: " *In the case of cars, more smoke...*"
> G: *If it is carbon monoxide , more* (you have), *more pollution*"
> R: " *More pollution*"
> G: " *The point is: smoke could be smoke of anything. It is not necessary to be that black smoke... It could be smoke of sulphuric gas, smoke of oxygen, smoke of hydrogen, smoke of carbon monoxide*"
> (Ex. 1)

Although they were considering different types and qualities of smoke, some time later, when we were working with a model about employment (Task 1 - Activity 2), they made some comments about what should be taken into account to construct a model:

> G: " (..) *We have to think in a more general way. Because in the way I was thinking we were particularising. But we can not particularise, we have to be on a general level. So, on a general level, jobs offering is more or less the same*"
> (Ex. 2)

In the next model we were talking about life expectancy. It was affected by *quality of life*, which was, by its turn, affected by *health.*

> R: " *It is because it is a model, isn't it? We can't imagine that a guy who is healthy can work and do other things. He can improve his quality of life constantly. He can be constantly improving but we can't consider this in*

the model, because you have to abstract, take into account only the factor (element) *health and life expectancy, don't you ? So it is something..."*
(Ex. 3)

I think it is not simple to talk about this subject here or at least to state, from what was observed, that it is very hard for the students to think abstractly when creating a model with LInkIt.

If we change the scenario and think about a (real) situation where someone - a person with some expertise in modelling - is trying to construct a model, he/she has clearly in mind beforehand what questions he/she is trying to answer through the use of the model. So, in this situation, that person can easily begin to construct a model from a very abstract point of view and, as things evolve, he could consider the possibility of adding more detail about variables or events which arise. But the scenario in this study is quite different mainly because the students do not have a clear idea of what they are doing. In other words, they do not have questions to be answered. They are talking about some ideas like pollution, unemployment, etc. and using the software to externalise them. Therefore I think that from this point of view it is relevant to consider all aspects of a certain problem.

It is true to say, from the many passages presented in the five analyses of this study, that they tend to use concrete examples to justify their ideas. But what is more important here is that they could use LinkIt to represent and test some hypotheses about these ideas.

USING THE GRAPH OPTION

Like the other groups this option was not very much explored. We used it more in the beginning of the experiment to see the differences between 'gradual' and 'immediate' variables. But they also used it to explain certain behaviour of the model. One example arose during the construction of the model about money, where Gabriel decided to plot a graph of *money* and used it to explain the interdependence between *money* and *wealth* (See also Example 26 in the topic Feedback Loop).

Here they had already constructed a negative feedback loop between the two variables.

F: " *Graph of what do you want ? Money ?"*
G: " *Money. Graph of money"*

G: " (running the model) *You see. It stays at a certain height until wealth arrives there and then it begins to go up, in a proportional way*"
(Ex. 4)

An interesting thing was that they used mathematical language like "*it is a crescent* (i.e. growing) *function*" to describe the graph of a 'gradual' variable.

BEHAVIOUR OF THE MODEL WHEN INITIAL VALUES ARE CHANGED

It was clear to them that the fact of a causal variable being constant does not mean that the dependent variable should be constant (This situation also happened with Group B). In the passage below we were talking about pollution and the variable *pollution* was gradual.

F: " (...) *let's set just one* (of the causal variables) *here and see what happens. If we have just one, pollution has to grow , hasn't it ?*"
R: " *I think it grows but less than if there were both, doesn't it ?*"
F: " *What do you mean by less there ? You mean it doesn't arrive at the top or it increases more slowly ?* (...)"
R: "*Only with the smoke of cars it* (*pollution*) *will increase more slowly* "
(Ex. 5)

GRADUAL AND IMMEDIATE VARIABLES

Like the other groups they tended to associate the idea of being 'gradual' to something that increases little by little and 'immediate' to something that goes straight to a different position.
There was an interesting situation during Task 3 when they were criticising the model about water-cycle. Although all variables were 'immediate' when we ran the model they saw them moving between different positions (because of the feedback loop) which was sufficient for Renata to think they were gradual:

F: " *In this case there, is it* ('sun shining') *gradual or immediate there ?*"
G: " *It is immediate* "
R: " *Immediate ? Immediate is when it sticks at once, it stay steady there*"
G: " *Immediate is when it goes up straight*"
R: " *But it* (sun shining) *goes up and down*"
(...)

F: " *What do you think Renata ?*" (asking about the behaviour of 'sun shinning')
R: " *I think that when it is immediate it jumps from one extreme to the other. It jumps from one* (position) *to the other and stops. When it is gradual it decreases* (little by little) *until it arrive at the same level, doesn't it ?*"
(Ex. 6)

Again by the end of the work with this model I asked them " (...) *why do you think that they* (the variables) *have to be immediate ?*" and their answer was based on a combination of the behaviour of the variables on the screen and how an ecosystem works.

R: " *Any change... for instance...we should think from an ecosystem perspective. Any change in the ecosystem will immediately be reflected in something on it* (the ecosystem). *So anything done to this model should immediately be reflected on it* (...)"
(Ex. 7)

Much later, during the work with Nomad population (Task 3) they addressed 'immediate' variables using terms like *instantaneous* and *quick* .
It seems that the way a variable should behave on the screen is a sufficient condition for the student to determine its type. This behaviour, sometimes also seems to happen when they are trying to "fix" a model (See also topic: "Working with the Model About Nomad Population"): They change the type of a variable, without paying attention to what it is representing, just to see if the model runs well:

G: "*The temperature should be gradual...*"
R: *But isn't it* (gradual) *?*"
G: "*No. It is immediate*"
F: " *Temperature of the earth is now gradual*"
(After running the model and not satisfied with its behaviour...)
G: " (...) *this temperature of the earth is gradual. I will change it again to immediate*"
R: " *Yes it didn't make too much difference*"
G: " *Yes. It didn't make any difference*"
(Ex. 8)

In addition to these common patterns of behaviour they also gave some different explanations to these variables, which were closer to "system-definition": They associated 'gradual' variables to something cumulative and the 'immediate' variable to something proportional to its cause. In the next two topics I try to show some examples of these views.

SEEING GRADUAL VARIABLES AS CUMULATIVE

During the construction of the model about pollution they suggested two variables: *chemicals* and *smoke of the cars*. When we were about to connect them to *pollution*, they told me that "*they are* (*chemicals* and *smoke of the cars*) *two factors that are going to accumulate forever*". In this first attempt *pollution* was 'immediate' and after running the model I started making some questions about its behaviour:

> F: " (...) *pollution, how does it function in real life ? Is it something that goes... changes instantaneously or is it something that grows little by little ?*"
> G: " *It goes little by little*"
> R: " *Yes, because it is cumulative isn't it ? So every time more...*"
> (Ex. 9)

SEEING IMMEDIATE VARIABLES AS PROPORTIONAL VALUES

In the passages below we were working with the model about unemployment. Basically this model had two variables *unemployment* and *hunger* (the dependent variable). In the beginning *hunger* was 'gradual' and before they ran the model I asked them what behaviour they would expect from *hunger.*

> F: " *In this case here, how do you... the type of the variable hunger there, the type of unemployment ? How does it evolve as time goes on?*"
> G: " *It is proportional*"
> F: " *What does it means, "proportional" ?* "
> R: "*You have as much unemployment as you have hungry*" (the number of unemployed people is the number of hungry people)
> G: *Yes. If X people are unemployed then X people are hungry*"
> (Ex. 10)

After changing the type of *hunger* to 'immediate' and running the model I tried to confirm their ideas (See Figure C.1):

F: "*OK. So unemployment and hunger. I mean, for you hunger is something...*"

R: " *It is the direct reflex* (consequence) *of unemployment.*"

(Ex. 11)

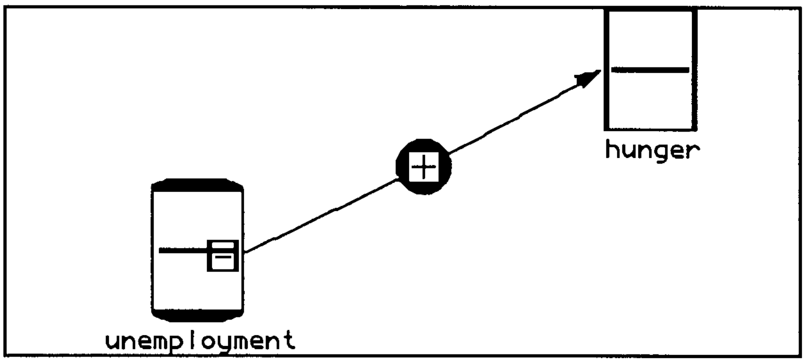


Figure C.1: Second model about Unemployment

In the next passage we were talking about the model of life expectancy (Task 1 Activity 2). It was caused by *quality of life* (gradual), which was, by its turn, caused by *health*. They wanted *quality of life* immediate. In other words proportional to *health*. Again they explained the idea of being proportional:

G: " (...) *if I increase health by 2* X , even if it is starting from zero and quality of life is not starting from zero, I mean, suppose it is equal to 2*X , that quality of life should increase 2*X above what it is* (already). *It will be 4*X, do you understand ?. It increased proportionally. If one side increases, the other one has to increase by the same*"

(....)

G: " *We have to put* (change) *quality of life direct. Immediate.*"

(Ex. 12)

RATE OF CHANGE

It seems that in some situations the students are seeing, in a qualitative way, that the causal variable is a rate of change of its dependent one. This is shown in Example 5 above when they noticed that the speed of the level of *pollution* is not the same when you have a different number of causes.

The idea is clearer in the passage below where they were constructing a model about money (Task 1 - Activity 2) and both variables 'money' and 'wealthy' were connected by a positive feedback loop (See Figure C.2).

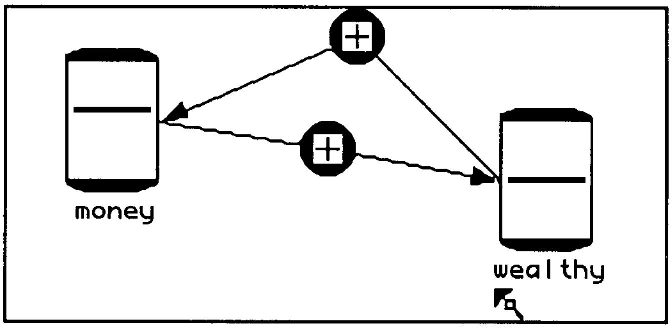


Figure C.2: Model about Money with a feedback loop

F: " *So, you connected money to wealthy and wealthy to money ?*"
G: " *When we changed the level* (of money), *we, what we put here in the level was the factor of increasing and not the quantity of money. If you put a little here* (set the level of money), *it will be how much it will increase. The proportion that it will increase. It will increase just a little. If I put it in the maximum, it will increase very fast.*"
(Ex. 13)

INTERPRETING THE LEVEL

It seems that in the beginning (Task 1 - Activity 1) they were interpreting differently the position of the level. Gabriel was associating the idea of the level in the middle with zero. He spontaneously told me "*This level there is what ? In the middle it means 'it doesn't have'* (zero) *?*". But Renata associated it with something normal (meaning that there exists some pollution but it is not harmful): " (...) *it would be considered as the acceptable level of pollution. It would be in the middle*". Thus they predicted in a different way what would happen to *pollution* if we set *smoke of the cars* high and *chemicals* at the bottom:

F: " (...) *in this case here what do you think is going to happen* (to pollution) *?*"
R: " *Pollution will increase but more slowly*"
G: " *It will stay the same. It will be in the middle*"
(Ex. 14)

What was also interesting here was that after running and seeing *pollution* not increasing, Renata gave a good explanation of the situation:

R: " *If someone considers it* (the level in the middle) *as the acceptable level of pollution then it could be what we expect because if you don't have any contribution from chemicals, it could be that the pollution that comes only from the cars would be acceptable, do you understand ?* (...)" (the smoke that comes from cars is not sufficient by itself to raise the level of pollution)
(Ex. 15)

A bit later she began to consider the other possibility:

R: " *It could be two things: if ithe middleis considered as zero it will continue in the middle. But if ihe middle is considered as the acceptable limit, it will decrease. What we are going to consider ?*"
(Ex. 16)

Later, during the improvement of the model about pollution, they were considering the possibility of changing the range of *health* to "bigger than zero". It seems that they were associating, in a qualitative way, the value zero to the middle of a box .

G: " *But I think that health has a minimum value.*"
F: " *It has a minimum value ?*"
G: " *The health of the population is going to diminish until a certain point. After that point the person dies and health finishes. So we had to put that factor that put the minimum value* (...)"
F: " *That factor that you are talking about is there...in that big box of options* (Information box) *?*"
G: "*Yes the range must be bigger than zero*"
R: " *It has to be different from zero. Is that you want to say ?*"
G: " *No. It has to be bigger than zero* (...)"
(Ex. 17)

But during the work with the water-cycle model (Task 2) they associated the bottom of a box with zero and its middle with something normal:

F: " *What is the meaning of clouds in the sky there in the middle for you ?*"
R: " *It means there are some clouds in the sky*"
(...)

F: " *What do clouds in the sky there in the bottom mean for you ?*"

G: " *There at the bottom ? There are no clouds*"

F: " *OK, and sun in the middle ?*"

G: " *Sun in the middle means it is normal*"

R: " *It is a sunny day but not too hot*"

(Ex. 18)

INPUT COMBINATION

When we were working with the model about pollution (Task 1 - Activity 1), they mentioned how the causal variable was calculating its new value (Gabriel said it was calculating the "average"). So I decided to argue with them about the input combination. In the passage below the Information Box was opened and the input combination of *pollution* was set to 'average':

F: " *How do you think they are calculating here the input combination ?*"

G: " *Oh, yes. They're calculating the average, like I said*"

F: " (...) *It is doing the average. There exist two other options: need all or need any. In this situation here. What do you think would be a better input combination than average?*"

R: " *All factors* (cause). *Chemicals as well pollution* (*smoke*) *of cars* (are needed)"

F: " *It would be need all or need any ?*

G: " *Need any*"

R: " *I don't understand it* (...) *what would be input combination anyway ? What would it be ?*"

(Ex. 19)

It seems that Renata did not understand the idea of input combination and her choice - 'need all' - was based on the fact that both variables have to be considered in order to calculate the resultant *pollution*. In fact a little bit later she said: "*But we are working with two factors, if you neglect the chemicals, you will be only working with smoke of cars* (...)". After that we ran the model using 'need any' and she got the idea: "(...) *it is the quantity that enters pollution. It is the factor that is being considered to* (calculate/cause) *pollution. How the calculation is done*"

'GONOGO' VARIABLES

The necessity of using 'GONOGO' variables came during the improvement of the model about pollution (Task 1 - Activity 4). They decided to connect *mortality* to *chemicals of industries* and to *smoke of the cars* to make them go down. But when they were doing it they discussed about a possible wrong interpretation that could put on what was going on:

> R: " *But you have to take care, because, it may happen that when you see this model you can think: 'Ah ! more people dying then the industries will produce less, pollute less.' Which is not true.*"
> G: " *That is what I am trying to say. The mortality can exist, it can increase, increase, but it arrives at a level where it begins to influence the chemicals of industries and smoke of the cars*"
> R: " *Isn't it better to put an intermediary factor like pressure of the population ?* "
> (...)
> F: " (...) *Is it something that after a certain level...?*"
> G: " *Yes it is a matter of a level. It arrives at a certain level that is not possible anymore* (to accept) *so the pressure begins.* (...) *the pressure* (*of the population*) *is neither gradual nor immediate, it simply exists or not* "
> (Ex. 20)

From the passage above it seems that they wanted a variable that would act for a certain period of time and after that stops (something like a threshold). But according to the implementation of the system the 'GONOGO' variable only stops functioning if the result of its inputs makes its value smaller than the value represented by its level. Otherwise it stays 'ON'.

After this passage I thought it was interesting to present them the idea of 'GONOGO' variable changing the type of *pressure of the population.* I told them what was the idea of the variable and how it works. After that we ran the model with different initial values and it seemed they got the idea. But when we were about to finish the task I asked them about the variable to confirm their ideas and Renata gave an unexpected (for me) interpretation (See Figure C.3):

> F: " (..) *what are you thinking there* (about the GONOGO) ?"
> R: (..) *if the popular pressure is big* (when the level is near to the top), (then) *even if the mortality is small it* (*popular pressure*) *will activate.*"
> (...)
> F: " (...) *what did you say ?*"

R: " *If the popular pressure is small, even though the mortality is big the pressure will take a long time to begin to happen*"
F: " *You are saying that if the popular pressure is down there, you are saying...*"
R: " (I am saying) *that mortality could be very high, it has to be very high in order for popular pressure to begin to happen* (...)"
(Ex. 21)

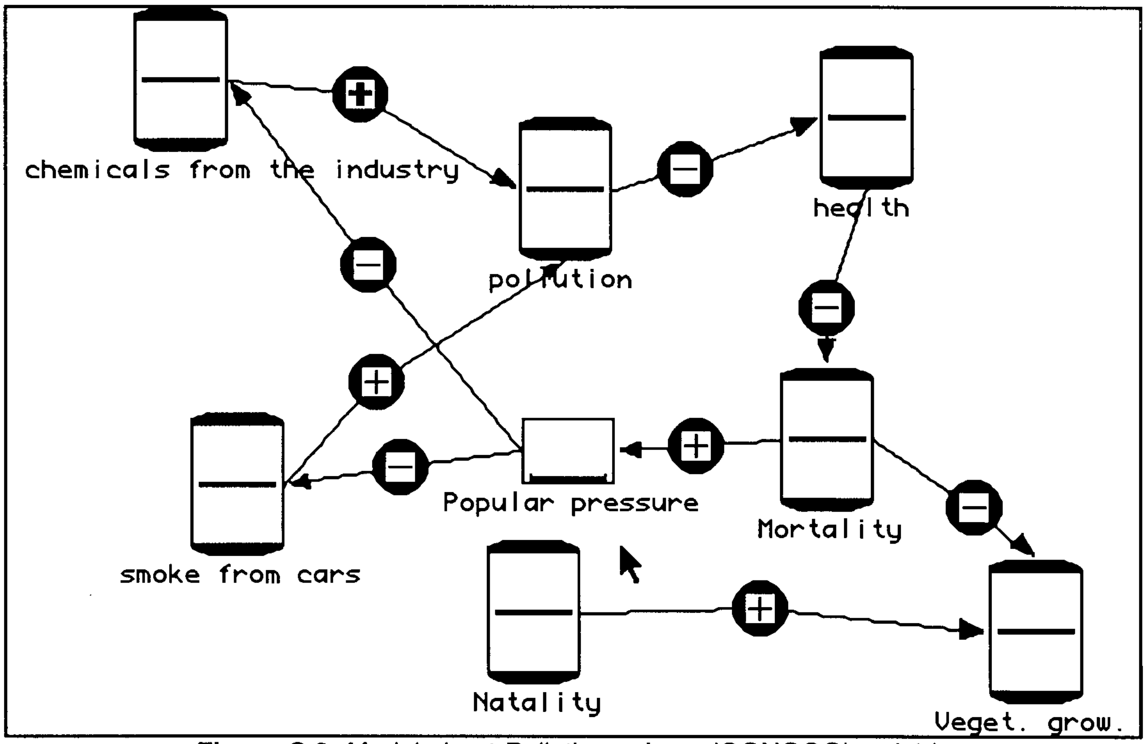


Figure C.3: Model about Pollution using a 'GONOGO' variable

It is clear from the passage above that she was giving another meaning to the level of 'GONOGO'. It seems that for her the level represents how powerful *popular pressure* is and not, according to Gabriel's words, "*the limit of growth of mortality*".

One possible explanation for this line of thought could be that she was trying to transfer to this new kind of variable her previous knowledge about what is represented by the levels of the 'gradual' and 'immediate' ones.

After that I explained again how it works using some different possibilities with the model and asking them to explain to each other what they understandood and what they would expect to see. Some time later Renata told me:

R: " *Then there* (the level of *pressure of the population*) *is how much pressure the population can accept, how much mortality they accept* (before) *taking an action. To change something.* (...) *I was analysing the pressure of the population in relation to the industries. The pressure of the population would be bigger if the mortality was bigger, do you understand ? Which was not correct, it is the limit of pressure that the population can accept, it is the contrary, it is related to the population and not to the industries .*"
(Ex. 22)

At this point I became convinced by her arguments and we moved to the next task.
Another interesting thing here was a suggestion given by Gabriel about how a 'GONOGO' variable should be represented on the screen:

G: " *It is like you had taken something and split it into two* (variables). *I see it like this: the quantity of demonstration inside mortality* (works) *like a limit. For instance, mortality is here at this level, OK. ? The quantity of manifestation is above it. Then mortality is going up, going up, going up and reaches it. And when it reaches the quantity of manifestation, the box turns on and it begins to function. But you instead of putting this level inside mortality, you separate it and put it outside* (in another variable) (...)"
(Ex. 23)

During Task 2, when we were working with the model about the water-cycle, it seems that Renata got the "right" way of thinking about 'GONOGO' variables (but wasnot quite convinced about them) and Gabriel was still stuck at the idea of the variable being a threshold.

In this passage we were discussing the interpretation of the level of *temperature on the earth* and Renata began to think that *evaporation* (which is caused by *temperature of the earth*) should only happen after a certain value of the *temperature of the earth* is reached.

R: " *Shouldn't the temperature be a GONOGO ?*"
G: " *Do you want to make temperature GONOGO ?*"
F: "*Why* (do you want it) *GONOGO ?*"
G: " *I don't think so*"
R: " *Because after a certain limit it begins to evaporate*"

G: " *But in this case is the evaporation that has to be GONOGO. It is the evaporation that occurs or does not occur. The temperature varies*"
R: " *It is right. Evaporation should be GONOGO*"
(Ex. 24)

During Task 3 they considered the possibility of using a 'GONOGO' variable to represent a drought that happened every thirty years (according to the text) but they did not do it because, according to Gabriel " *there isn't a cause that could trigger it*" .

FEEDBACK LOOPS

The first use of a feedback loop was spontaneously suggested by them during the Task 1 - Activity 2, when they were working with the model about money (See Figure C.2 above). They wanted to represent the idea that: " *wealth brings* (causes) *more money and money brings* (causes) *more wealth* ". After creating it Renata justified it using a concrete example:

R: " *Take your money and invest it in FAF* (a kind of saving account in Brazil) *it will double. Then you buy a house. Then the house appreciate in value and then you're more and more rich* (...)"
(Ex. 25)

A bit later Gabriel gave an explanation based on its behaviour on the screen:

G: " (...) *you put this* (*amount of*) *money.* (But) *more money more wealth, then wealth will increase. But wealth increased but more wealth more money then* (*money*) *will increase a bit. It will increase a bit because wealth increased just a bit. Now the graph of wealth is much bigger, and bigger it is, more wealth. Then wealth will increase faster. But because wealth increases very fast, money has to increase fast as well and now money goes up there. And now because there is a lot of money , wealth also increases a lot and it arrives at a point where they* (the levels) *become equal* (same height at the same time). *And when they become equal they begin to increase together*"
(Ex. 26)

SIGNS OF LINKS

The idea of using a negative link came during the construction of the model about jobs on offer (Task 1 - Activity 2). Again here, like the other groups, they tended to think about the relationship between the variables onlyconsidering the possibility of the growth of the causal variable.

> G: " *More jobs offering you have...*"
> R: " *Less unemployment* (you have)"
> G: " *Yes, less unemployment*"
> R: "*But is there less in this ...*"
> G: " (...) *I think so* (...)"
> (Ex. 27)

Later when we were working again with the model about pollution they correctly chose the 'minus' sign to connect *health* and *mortality* but were slightly confused on how "to read" the relationship.

> G: " *Now lets connect them. More health* (you have)... *More health less, Er, more, wait. Less health* (you have) *more mortality, right? So it is minus. Inverse relation.* (...)"
> (Ex. 28)

IDEAS ABOUT MODELLING

Sometimes during the construction of the models they externalised their own ideas about modelling. They discussed how it functions and what it represents. In this section I will try to show some examples of these ideas.

The first is about the behaviour of the system. It seems that they could choose a certain way of interpreting what they were creating on the screen and the system would function in that way. One example of this behaviour is the Example 15 in the topic above "Interpreting the Level".

Another idea is that they could think about conditions that were not represented in the model but have some influence on it and yet sometimes use other variables to represent these conditions. In the three passages below we were talking about the model pollution of the air at three different moments:

> Gabriel giving a justification (after running the model) for not seeing *pollution* growing at the speed they wanted.

G: " *We also have to consider that there are some other factors that diminishes pollution like weather characteristics, wind, forest, open spaces. These are things that make pollution decrease*"
(Ex. 29)

Renata giving a justification (after running the model) for not seeing *pollution* growing.

F: " (...) *It is diminishing pollution, isn't it ? Is it* (this behaviour) *true or not true ?*"
R: " *It depends, it depends. In the model it could not be true. If you are not considering the air, the movement of the air. Then it is not true. Do you understand ? (...)*"
(Ex. 30)

The variable *pollution* was changed from 'bigger than zero' to 'any value' and after running the model again it functioned in the way they wanted. They constructed a justification to have it as 'any value' (See Figure C.3 above).

F: " (...) *In this case there where we are using pollution as any value, is it closer to what you..., to the reality ?*"
G: " *Yes. it is correct. Now we could interpret it in the following way: without* (the causes of) *pollution acting the tendency is for the pollution to diminish until it arrives at a point where there is no more pollution. I think that now is correct. It is like we were accepting the factors that diminish pollution. Factors that , in the case here are smoke of cars and chemicals of the industries* (variables that exist in the model) *but in reality would be weather factors, environment factors and things like that.*"
(Ex. 31)

A slight variation of the idea above is that they could permit certain things to exist in the model that they know do not exist in reality (See Figure C.3 above).

They ran the model about pollution and *health of the population* went below the middle.

R: " *You can't have negative* (number of) *people. Health is going down, going down* (and) *you will have negative* (number of) *people.*"
G: " *Yes can't have a negative number.*"

R: " *Ah, it is OK. It is a model. In a model we can have this.*"
(Ex. 32)

Despite this they knew that the model has to have a commitment, at least at a certain level, with something that happens in the real world.

We were working with the model about nomad population and Gabriel was about to change some links between the variables in a way that was not making sense to Renata.

G: " *Everything is connected here. It should be, from each one 3 arrows* (links) *should go out , one to each of the others and coming back. Everything is linked* (...)"
R: " *Tell me why. What is the logic* (in your thought) ?"
G: " *You will see what will happen*"
R: " *It doesn't matter what is going to happen to the* (behaviour of) *model, what matters is reality.*"
(Ex. 33)

The models constructed by them also served to think about the real world.

In the passage above we were talking about the model water-cycle. They had had evaporation as 'GONOGO' and after running the model they commented:

R: "*But also here evaporation happens every day, every time. Even on an awful day evaporation happens, doesn't it ?*"
G: " *Yes but I think like this: It arrives at a certain level that it doesn't influence, do you understand ?*"
F: " *It arrives at a level that it doesn't influence ?*"
G: " *Yes there is a level of evaporation that is so small that it is not going to increase the quantity of water on the Earth or the number of clouds in the sky. Something small.*(...) *but when it is very big it begins to function* (...)"
(Ex. 34)

TALKING ABOUT THE SYSTEM (POTENTIAL FORMALISATION)

Another thing that was noticed with this group was their ability to achieve some formalisation. They could analyse and think about the variables and their relations from a system perspective, abstracting the fact that they were representing a physical problem.

In the passages below we were talking about the model pollution of the air. The feedback loop had already been created.

G: "*See what I told you. It stopped, stopped. Nothing moves any more. Although smoke of the cars and chemicals of the industries have finished, the popular pressure continues.* "
R: "*Of course. So that it doesn't come back* " (in order that *pollution* doesn't increase again)
G: " *No. It is simply because pollution cannot be below zero* (...)"
(Ex. 35)

There was a suggestion to insert a 'GONOGO' variable (called *natural recovery*) in the model in order to make *health* of the population increase after the decreasing of *pollution*. But before creating it Gabriel realised it wouldn't function.

G: " (...) *We can't do it because in the first cycle pollution will be zero. Do you understand ?* (Then) *The mortality will be zero. Then health would increase anyway. But the natural recovery should only function in the second cycle when health would diminish to a certain level* (...) "
(Ex. 36)

Another two examples occurred during the work with the model about the water-cycle. In the first they were considering whether to connect *rain* directly to *clouds in the sky* and erase the link between *evaporation* and *rain*, but after thinking a little about it, Gabriel decided not to do it because, as he said, "*There exists an indirect link there* (between *clouds in the sky* and *rain*)".

The other example is an explanation of why they should change the type of *temperature of the earth* to immediate.

G: " *It is because when the evaporation is in the upper part or at the bottom, it has to wait for the sun to shine and the temperature of the earth , that is gradual, it has to wait for the sun go down gradually and then it* (the *sun*) *will make the temperature go down gradually as well. When it* (*temperature of the earth*) *is immediate no* (the situation is different). *You spend half of the time.*"
(Ex. 37)

FORMULATING HYPOTHESES

During the work with the model they basically formulated and went on to test two different types of hypothesis. The first, which happened much more frequently, was related to the idea of "making the model function correctly". A typical case happened when they were not seeing the model behaving properly and then they worked out a justification to change a parameter of a variable or of a link and went on to try it out.

The second case was that of using the model to make some predictions about the outside world.
In my opinion the reason this type of hypothesising did not happen very much, was that the way the tasks were prepared did not leave too much "space" for the students to think about them. Nevertheless some examples occurred.

> They were working with the model about life expectancy. There were five variables in this model: *money, health* and *living conditions* (quality of the place they live) causing *quality of life* which in turn caused *life expectancy* (See Figure C.4 below).

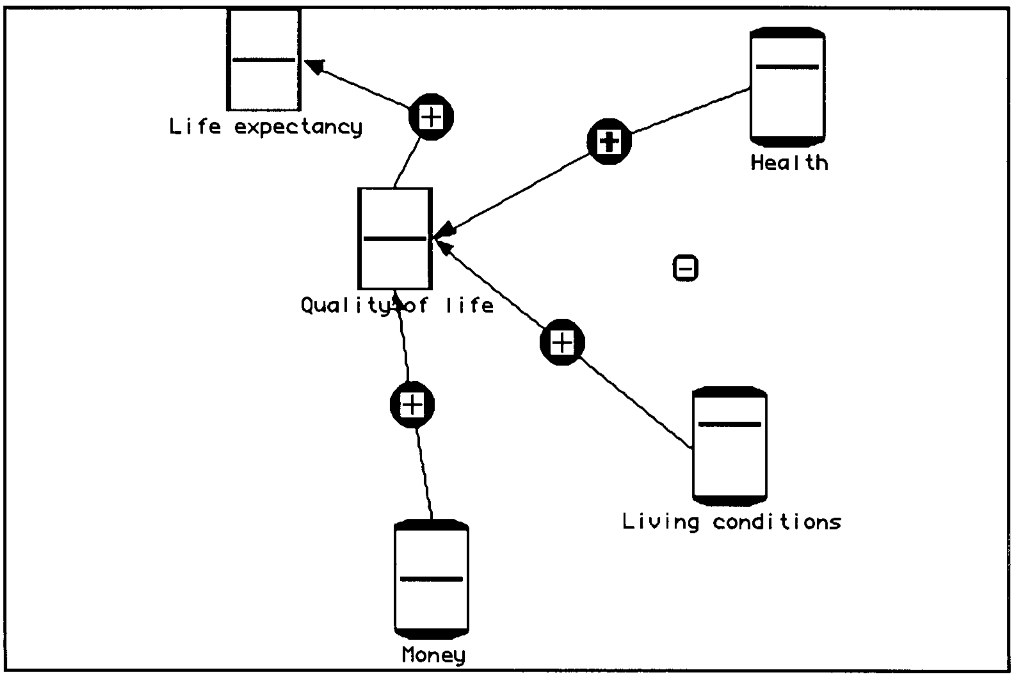


Figure C.4: Model about Life Expectancy

> F: "*So you created two more variables, didn't you ? Money and living conditions.*"
> R: " *And they influence* (cause) *quality of life*"
> F: " *Which influence quality of life...*"

R: " *Let's see if money is very important. Lets see if the person living in a good place and being healthy, if he will have a good life expectancy and quality of life* "
(Ex. 38)

WORKING WITH THE MODEL ABOUT A NOMAD POPULATION

Before they constructed the model, I told them to divide the problem into two parts. In the first they should try to construct a model before the arrival of the UN and their technologies. In the second one they should take into account the interference of the UN in the way of living of the nomads. After reading the text Renata reflected about it:

R: " *The text is showing that...nothing is better than the natural equilibrium, isn't it ? . You don't have to change things that are well in equilibrium, isn't it ? Although they have to struggle to live, everything was in equilibrium, their ecosystem was functioning.*"
(Ex. 39)

They based their model on the text they read. All the variables they created were mentioned there. Most of them represented quantities like *nomad population*, *herds* and *water*. But there were some that were used to represent an event like *moving* (to represent the idea that the population has to move to another area every 30 years) and others like *stock of water* and *vaccines* that were used as constants.

First they created some variables without paying attention to their types, and then they connected them.

In this model they created four small negative feedback loops (see Figure C.5 below) to represent the interrelation between the main variables *water*, *grazing fields*, *herds* and *nomad population*. The first one represented the relationship between *water* and *herds*. Its idea was much more apparent in the text and (maybe for this reason) it was even created before they ran the model.

G: " *More grazing fields you have bigger the number of herds, consequently less grazing fields*"
R: " (...) *the minus is stronger because the herds increases much more slowly than the grazing fields*"

G: " *How is it ? The grazing fields finishes quicker ? So this inverse relation* (negative relation) *is bigger* ."
R: " *Yes*"
(Ex. 40)

Contrary to the first one, the other three feedback loops were created a bit later as attempts to "fix" the model. They spent a long time discussing the relation between them specially because what they wanted to represent was not so easy to do with the tools available. For instance, one thing they were trying to represent was that when *grazing fields* increases, *herds* has to increase and when *herds* increases, *grazing fields* has to diminish. But they did not want to see *grazing fields* increasing when *herds* diminishes (because they were eaten by the nomads).

One possible explanation for this behaviour could be a wrong interpretation they were giving to the negative sign. If you think about it as something that makes a variable decrease then it does not make sense to think that when the cause (*herds* in the case) diminishes the dependent one (*grazing fields*) has to increase.

Also during this discussion about the behaviour of this feedback loop they suggested a new kind of variable. A variable that could only be smaller than a certain value set by the user:

R: " (...) *Why doesn't the grazing fields go down ?*"
G: " *Because the relationship functions in both directions. It should have another factor* (variable) *that says that grazing fields only decreases.*" (realising how the feedback loop between 'herds' and 'grazing fields functions)
R: " *So, put excess of consume*"
(...)
G: " *I have an idea. Lets put it* ('grazing fields') *smaller than zero* (...) *It is not exactly this that I want but...The ideal would be that you could choose a value and see it smaller than zero. It would be smaller than a certain value, doyou understand ?*"
F: " *No. Could you repeat it to me ?*"
G: " *For instance: there you have the options any value, bigger than zero and smaller than zero. One suggestion would be to put smaller than the value I would choose. Because what I would like to have is the level up there. The level of grazing fields would be up there and only decrease, it could not go up. It doesn't matter what...*"
(Ex. 41)

In my opinion the most difficult thing with this model was to think about the time interval we wanted to consider to analyse it. And because of this, the students were not sure about what they wanted to see. But, at least considering the model as a whole over a long time span, they were expecting something that should oscillate forever:

R: " *It* (the model) *is something that will never stop , it is something constant, this model can't stop. I mean, you activate it , you run it, it can't stop anymore.*"
F: " (...) *What do you want to see* (happening) *to the nomad population ? Do you want to see it in a normal position, going up and down... ?*"
R: " *Oscillating. Going up and down. Without having a big variation. And the herds also have to oscillate and grazing fields and water as well. All of them have to be oscillating...*"
(Ex. 42)

Also sometimes they did not agree about the period of time they wanted to observe:

R: " *But* (to have more) *water they have to move to find it* "
G: " *Forget it. This is the problem. I am thinking like this: before they move. Lets suppose they haven't moved* (yet)"
(Ex. 43)

Another interesting thing was the way they looked at the model. Every time they ran it, they tried to analyse the behaviour of the variables locally.

F: " *You put a lot of water and grazing fields...*"
G: " *Herds in the middle. In the middle no. A little bit up. I will run it...*"
(The model is running...)
G: " *It is diminishing the grazing fields and water...*"
F: " *Herds is going down, food is going down and population...*"
G: " *Water has to be immediate !. I will stop it.*"
(Ex. 44)

This kind of reasoning "forced" them to make a lot of modifications to the variables (changing their type) and to the links between them without considering what they were representing (This idea can best be demonstrated by comparing the different types of the same variables in the Figures C.5 and

C.6 below). In the following example we see them trying a change in the type of a variable just to see if the model functions better:

R: " *Why did you change water to gradual ?*"

G: " *No, no just a quick test. I know that water is not gradual but...if I set it gradual...wait.*"

(*model is running...*)

F: " *Nomad population is going down...*"

R: " *It is in the equilibrium now* (...)"

G: " *In the end it stops. It always stops.*"

(Ex. 45)

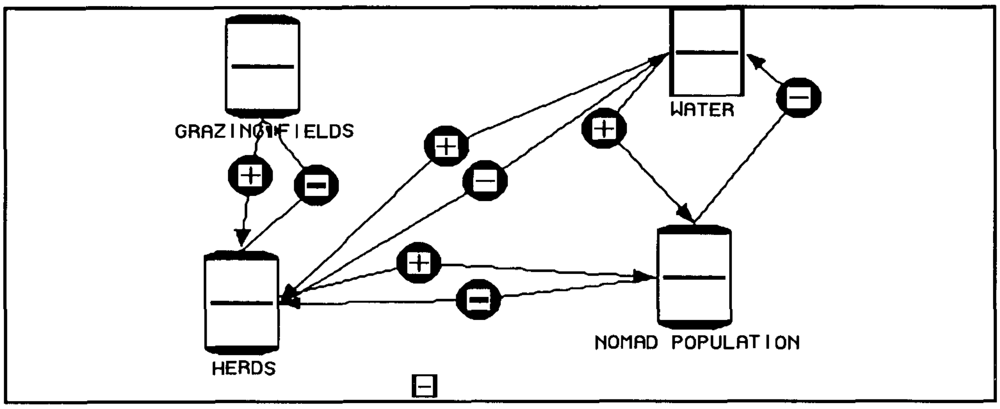


Figure C.5: Model about the Nomad Population featuring four feedback loops

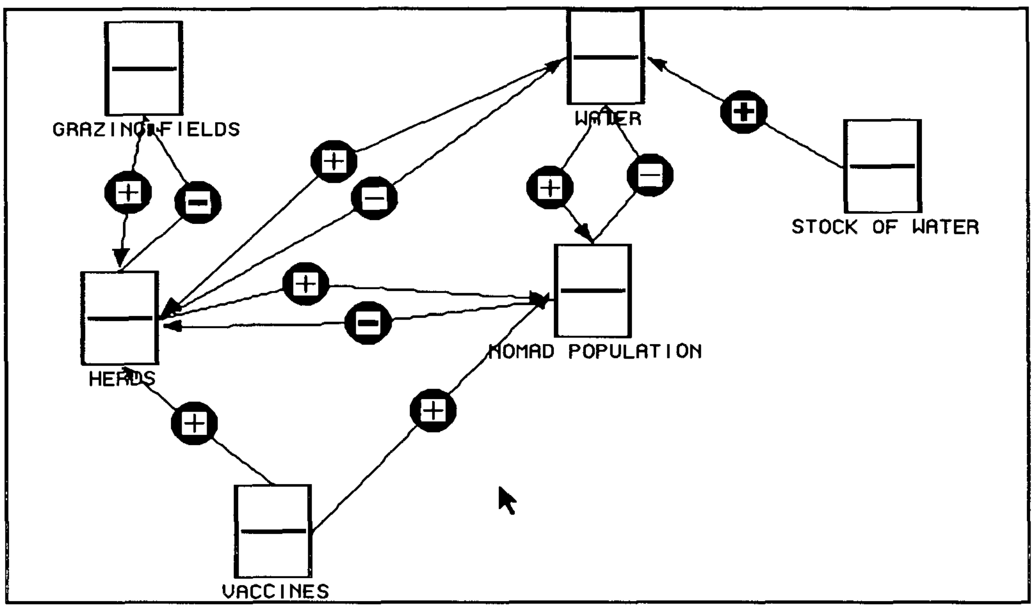


Figure C.6: Model about the Nomad Population after the arrival of the UN

INTERFACE

The students managed to use the interface. Despite this, there was a very big difference between them mostly because Gabriel was quite experienced with computers and Renata had never used one before. The main problem she had was to develop the motor-skills to control the mouse and perform operations like dragging, double-click and pointing in the right place. Some of these mistakes caused the system to interpret her actions in quite different ways , causing more problems.

Another thing was that Gabriel noticed that some menus in the menu bar were written in English and others in Portuguese. I explained to him that this was not a fault of the software but of the system because the cT language does not permit you to change the names of the menus “File” and “Options”.

APPENDIX D

CORE STUDY: SAMPLE OF A CASE STUDY

INTRODUCTION

This Appendix presents a sample of a case study related to the Core Study. The group presented here (Group E) is composed of two boys, Diego (13 years old) and Pedro (14 years old). Both were in the same school level.

They were very participative in the tasks. Specially during the learning phase they were ahead of me many times, making comments and questions about the way the system works.

During the writing of this case study (in the beginning of 1996) I tried as much as possible to follow a chronological description of what happened when they were using LinkIt II. Thus from it the reader can get a feeling for how the students knowledge of the system and thinking about models developed during the tasks.

Task 3 (Training Examples) - Activity 1 - Presenting "Go together links", direction of a link, add/subtract, any value variables

In this activity we first constructed a model about the total income of a waiter and after they suggested a similar example about depositing and withdrawing money from a banking account.

From the very beginning of this task they used semiquantitative thoughts to describe the relationship between variables. An example comes in the passage below where we were constructing the model about the waiter and talking about its relationships:

Fa: " (...) *and the relation between tips and final income ?*"
D: " *If the tips are bigger* (then) *final income* (will be) *as well* "

It seems that the idea of links being added was very intuitive for them. In the passage below we were talking about the consequences on *total income* when you vary *salary* and/or *tips* and they used some mathematical properties of addition to explain what happened to the model:

Fa: " (...) *if I put some salary and he doesn't receive any tips... look at his total income there it stays...*"
D: " *Stayed the same as salary*"
Fa: " (...) *now if I give some tips to him...*"
D: " *The final income is going to rise*"

Later they again made concrete some mathematical formalisms. In one case they associated the idea of zero with a situation where the person did not have any *total income* (this happened after introducing a variable called *expenses* that was connected to *total income* via an opposite link) and soonafter they associated the idea of negative numbers with the person borrowing/owing money.

They also noticed that the arrows in the middle of the links carried with them some information about the properties of the links. In one situation they commented that the direction of the arrows in the hot point of the links (go together links) was related to the direction of their causes and later they also noticed that the size of the arrows is related to the strength of the links.

A new interpretation of the way 'go together' links work emerged when they suggested the creation of a new variable called *discounts* (to represent compulsory discounts you have every month). Instead of connecting it to *total income* they suggested an opposite link going to *salary* (See Figure D.1):

Fa: " (...) *and now* (we have) *deductions from the salary, how does it determine final income ? Bigger the discount...*"

P: " *Smaller the total income* ."

D: *Isn't it better to connect it to salary ? ....* (to connect) *discount of the salary with salary instead of final income.* "

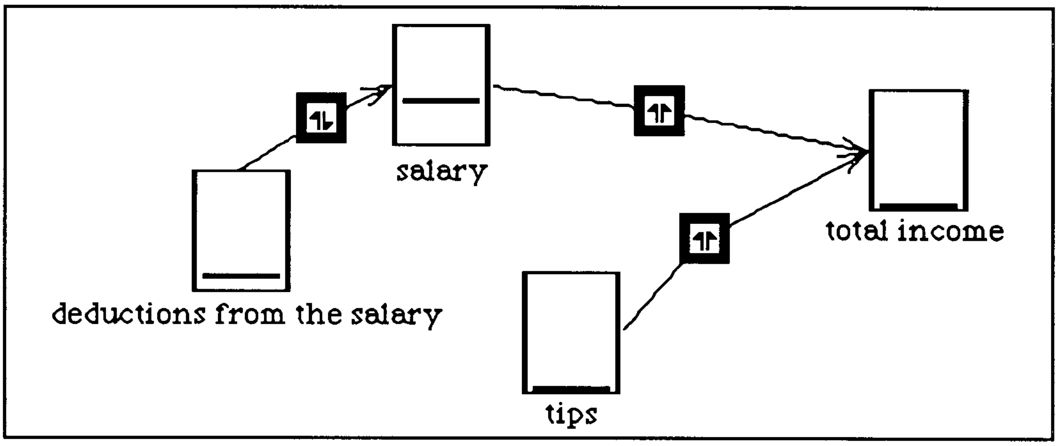


Figure D.1: Model they had in their minds about discounts and salary

It seems they are constructing a meaning to the relation where the net salary is the gross salary minus the *deductions.* In other words, *salary* is equal to the value it already has (when you set its level) minus the value of *deductions*. This idea works perfectly well in the real world but 'go together' links do not permit this kind of construction.

Task 3 (Training Examples) - Activity 2 - Exploring a model about Accidents on the road

Again here they noticed some different aspects of the links. After I gave them a brief description of the variables in the model, they said "*this* (link) *here is also bigger*" trying to point out that the link between *dangerous driver* and *total number of accidents* is 'strong'.

As in the other groups the idea of calculating average is related to getting a result in the middle of certain factors. In the passage below I was trying to make them perceive the input combination between *bad quality of the roads* and *dangerous driver* and they came up with an example where 'average' could be used:

D: " (...) *if we could increase bad quality of the roads and good quality of the roads...and then I would create an arrow to roads and would calculate the average between these two* (bad and good quality of the roads) *and we could do dangerous drivers...intelligent* (conscious) *drivers and connect them to*

drivers and calculate the average between them...then connect these two (roads and drivers) *to total number of accidents and accidents with death*"
P: "*Yes...*"

After that I explained the idea of 'average' changing the combination of *bad quality of the roads* and *dangerous driver* together with some simulations. Soon after they used this idea to combine use of *safety belts* and *total number of accidents* causing *accidents with death.* The argument they used was "*Because it can have many accidents..and with the use of belts, it may not have deaths...*".

Later when I asked them to insert a new variable in the model - *use of safety belts* - they decided to connect it to *dangerous driver.* Their justification was that "*more safety belts are used, less dangerous are the drivers*". But when they went on to do some simulations they were surprised with what happened to *dangerous driver.* Whatever initial condition they gave to *use of safety belts, dangerous driver* went straight to zero. The passage below is an evidence that they saw the relation between these variables as inverse:

D: " *Then it is wrong* (the model)... *because if use of safety belt is small... then the drivers...*"
P: " *Are more dangerous...*"

It happened once, during these simulations, that they set an initial value to *dangerous driver* and left *use of safety belts* equal to zero. When they ran the model, they were puzzled when *dangerous driver* became zero. Their idea was that if *safety belt* is zero then nothing goes to *dangerous driver.*

All of this "confusion" can be explained if you see a 'go together' link as a special case of 'cumulative' links where the value of the dependent variable is only calculated once: it adds to its initial value the resultant value of its inputs.

Task 3 (Training Examples) - Activity 3 - Presenting "How much change links", direction of a link, add/subtract

They seemed to understand the problem about the volume of water in the tub. After I showed them the model running with 'cumulative' links they started talking about the speed of increasing of the volume of water in the tub and what would hapeen to it if they changed the value of *tap.* They also devised some simulations to show how long it would take to fill with different initial conditions of *tap* and *drain.*

The similar example they suggested was about inflating a soccer ball that had a puncture (See Figure D.2). According to the size of the puncture it would or would not be possible to inflate the ball.

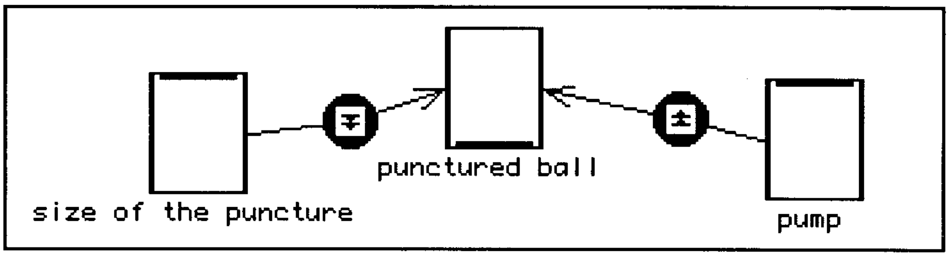


Figure D.2: Model about a punctured ball

One interesting point during this discussion was that they started talking about two other important aspects of the problem: the force you have to exert on the pump to inflate the ball and the pressure. They also discussed about how to represent these ideas in the model:

> D: " (...) *the pump would be a square* (variable) *that would alternate between upwards and downwards... and every time you make it upwards the ball would inflate, when you made it downwards the ball would stay stable* (no change on its volume) *and it would only empty because of this... the puncture* "

It is clear from the passage above that they were thinking about *pump* as an On/Off variable.

Task 3 (Training Examples) - Activity 4 - Explore a very simple model about Parking place

After I loaded the model and told them that they had to complete it by creating the links, the first thing they said was that the model "*looks like that one... (about) the volume of water in the tub*". It was clear for me that they were recognizing the similarity in the underlying structure of the models. However when they connected the variables, they made a link between *number of cars arriving* and *number of cars in the parking place* ('cumulative-same direction') and another one connecting *number of cars in the parking place* to *number of cars leaving* ('cumulative-opposite direction'). From our discussion of these connections, it seemed they were seeing it in two different ways. One interpretation they had was about the flow of cars from one place to another. The other interpretation was that they were seeing two events separated in the same model. One event was about cars coming and filling the parking place and the other one about number of cars in the parking place and cars leaving. In this second case, at least for Diego, the opposite direction link was not making sense. Actually, after some simulations he was the person who realised that the interpretation of the model was wrong saying " *No ! The right way would be the number of cars leaving diminishing the number of cars in the parking place, the number of cars arriving...*"

Task 3 (Training Examples) - Activity 5 - Presenting the combination multiplication and inverse

I introduced them to this task by showing them a model to calculate the price of a certain quantity of meat being bought. After the construction of the model, I asked them what kind of input combination should be used and they suggested 'average' and 'multiplication' . I then told them that the combination was set to 'add' and asked what could be done to show that 'add' was not appropriate. Diego said: "(...) *set price per kilo up there and number of kilos zero, it* (the result) *is going to be wrong*". After seeing the model confirm their prediction, they changed it to 'multiplication' and made the same test. This time the result was in accordance with their expectation and they accepted it as the right combination.

After I asked them for a similar example they suggested a model to calculate the price you have to pay for a certain number of boxes of disks. When they were creating the model I asked them what should be the type of the links. They said it should be 'go together' " *Because it doesn't change the speed* ". My interpretation here is that, for them, the causes do not have anything to do with the speed of variation of the level of *total price paid* (dependent variable).

The next model I showed them (about clouds and sun shining) was to introduce the idea of 'inverse'. The model we created had a feedback loop between *clouds* and *sun shining* but before I explained to them the idea of inverse we put the link between *sun* and *clouds* to sleep. When we turned it on and ran the model again, they were impressed with theoscillation of the model. I then explained to them why it was working that way.

Task 3 (Training Examples) - Activity 6 - Completing some models

The first model they worked with was about life expectancy. They connected *life conditions* to *health* and both to *life expectancy.* Although they kept the links as 'go together' they considered the possibility of connecting *life conditions* and *health* via a 'cumulative' one. They commented on it saying:

P: " *Because when life conditions increase, health will increase as well*"
D: " *No... life conditions increase, the health increases, it is correct... it changes the speed*"

It seems that for Diego the real justification was that in some way *life conditions* had to change the speed of variation of *health*, therefore the link had to be 'cumulative'.

After running the model and seeing all variables with the same final value no matter the initial conditions they set, they abandoned the investigation of this idea . They then put the link between *life conditions* and *life expectancy* to sleep (See Figure D.3) and ran the model again. This time Diego went through the model as though he was running it manually (he said: " *If life conditions is high... health of the population is high...if health of the population is high, life expectancy is going to be high as well*") and was satisified with what he said. It seemed here that the fact the output of the simulation was reasonable and coincided with his verbalization of what was going on there made him accept the model as "correct".

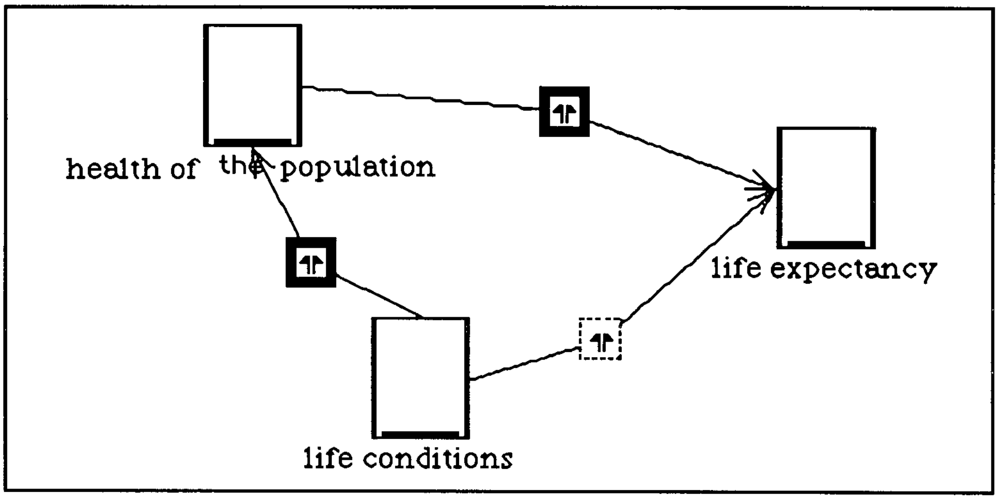


Figure D.3: Model about Life expectancy

The work with the other two models was left for the next meeting.

The next model they worked with was about car consumption. They first connected *car fuel comsumption* to *how economical is the driver* ('go together-opposite direction' link) and justified it saying " *It depends on how much fuel the car used to know whether the driver is economical or not* ". My impression here is that the relation is not only representing a cause-effect idea but also a way of calculating something.

It seemed that the way they were seeing the model was divided into two parts. The first one being about *distance travelled* and *car fuel consumption* (they connected them via a 'go together-same direction' link) and the second part about *car fuel comsumption* and *how economical is the driver.* Actually when I asked them whether *distance travelled* influenced *how economical is the driver,* they said "no", suggesting that there was not a link between them in the model. Also when they ran the model and disliked what they saw on the screen, they first set out to solve the problem about the relation between *distance travelled* and *car fuel comsumption* (they were ending at the same height) and afterwards they focused on the problem about *how economical is the driver* ending equal to zero.

To fix the first problem Pedro suggested changing the link to 'average' but Diego decided to change it to 'cumulative'. I asked them about their decision and they just said " *It has to be tested ".* But after running the model again they did not like to see *car fuel comsumption* increasing forever and said "*it doesn't have anything to do* (with what they wanted)". They then changed back the link to 'go together' and made it 'weak'.

When they were satisfied with what they saw, they moved to concentrate on the variable *how economical is the driver.* One interesting point here was that they began to recognize that it was dependent on *distance travelled* , though in an "*indirect way*". They first thought about changing the link to 'multiplication'. After that I began to make them think about a concrete problem and how we calculate the car fuel comsumption and how we determine whether a driver is economical or not. After this conversation they began to see that *how economical is the driver* should also be related to *distance travelled* and made some suggestions on how to change the model. Because we were spending too much time on this problem, I decided to move to the next one about food intake and weight.

In this new model they suggested connecting *food intake* and *energy spent* to *weight* using 'average' and 'go together' links. One interesting point here was that they were interpreting the model in different ways and therefore the type of the links should be different. For Diego they should be 'cumulative' because he was seeing things with days passing. Pedro was seeing it as an event that happens in the same way everyday, therefore it should be 'go together'. After some discussion both accepted each other view and agreed that with 'cumulative' links the model would be "*more practical*".

Task 3 (Training Examples) - Activity 7 - Presenting On/Off variables

After I introduced them to the idea of some events being controlled by other events, we went on to construct a model about feeling overweight and doing gymnastics. Figure D.4 shows the first model created.

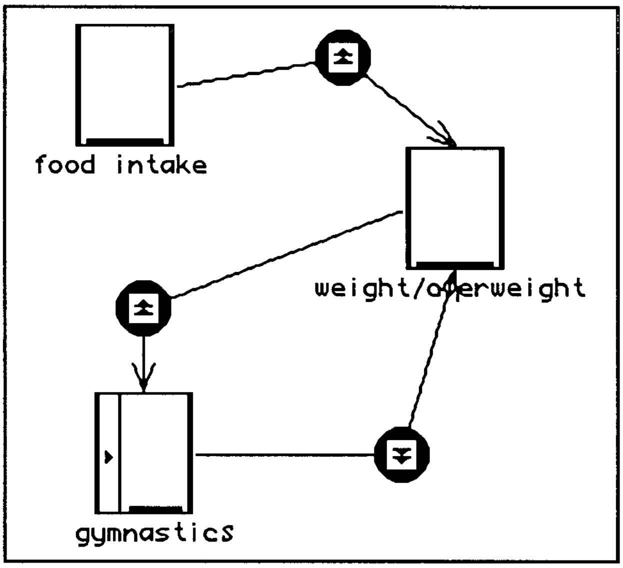


Figure D.4: First model they did using On/Off variables

They decided to use a 'cumulative' link between *weight/overweight* and *gymnastic* because "*you have to do gymnastics every day*". When they ran the model, they noticed that after *gymnastics* went up it never diminished again. They were not happy with it and decided to fix the problem. Because they were also seeing *gymnastics* as an On/Off event they decided to change its type, making it 'On/Off'. After changing it and not seeing the model work in the way they wanted, they began to associate the idea of doing *gymnastics* and eating more food:

> D: " *I would connect gymnastics to food intake... then when the* (weight of the)*guy goes above the trigger of gymnastics his weight would go down.... but because he would be eating... the weight would go up again... then he would do gymnastics..then when gymnastics was good... when it bumped the threshold...*"
>
> ...
>
> D: " *The threshold would represent the guy tired of doing gymnastics... when it went above... the guy began to do gymnastics ... when he began to do gymnastics he would begin to eat more...*"
>
> P: " *So he would continue eating the same amount...* (and) *doing gymnastics...*"

What I saw from the passage above was that they were trying to find a way to go around the problem. One possibility was to avoid making him (the guy) stop doing gymnastics.

This behaviour raises some issues about the understanding/use of 'On/Off' variables. The most important one is about the need to introduce the students to the subject not only by talking about their general functionalities but also by discussing in more depth all their properties and how they work with other components of the system.

Another interesting point in the passage above is the way they were seeing the model. Although in this newly suggested model there were not any independent variables, they started the process from the person beginning to do gymnastics. However, they were able to perceive and describe the feedback loop created.

The idea of the understanding and use of feedback loops became more evident during the similar model they suggested. It was about consuming energy and the bill paid. They first created these two variables and connected them. But while we were talking about them they said that *consumption of energy* should also be connected to *bill* (See Figure D.5) and explained why:

> P: " (...) *when energy arrives at this level here* (trigger level of *bill*), *he would stop consuming so much energy.... Yes, it would diminish the consumption of energy. So it would be necessary to make consumption of energy to there* (a link from consumption of energy to bill) *and from here to there as well* (a link from *bill* to *consumption of energy*)...(...) *If I leave just one link it is not going to diminish here* (*consumption of energy*)"

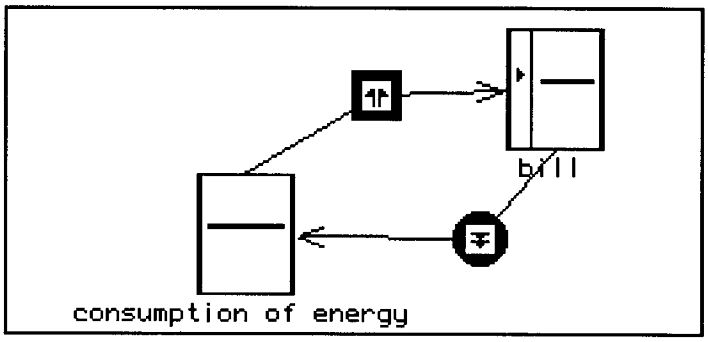


Figure D.5: Model they suggested about consumptiom of energy

Task 4 - Express their Ideas about a Problem - Migration to big cities

I was impressed by the first model they created because it used relevant variables that were connected in a very reasonable way (See Figure D.6).

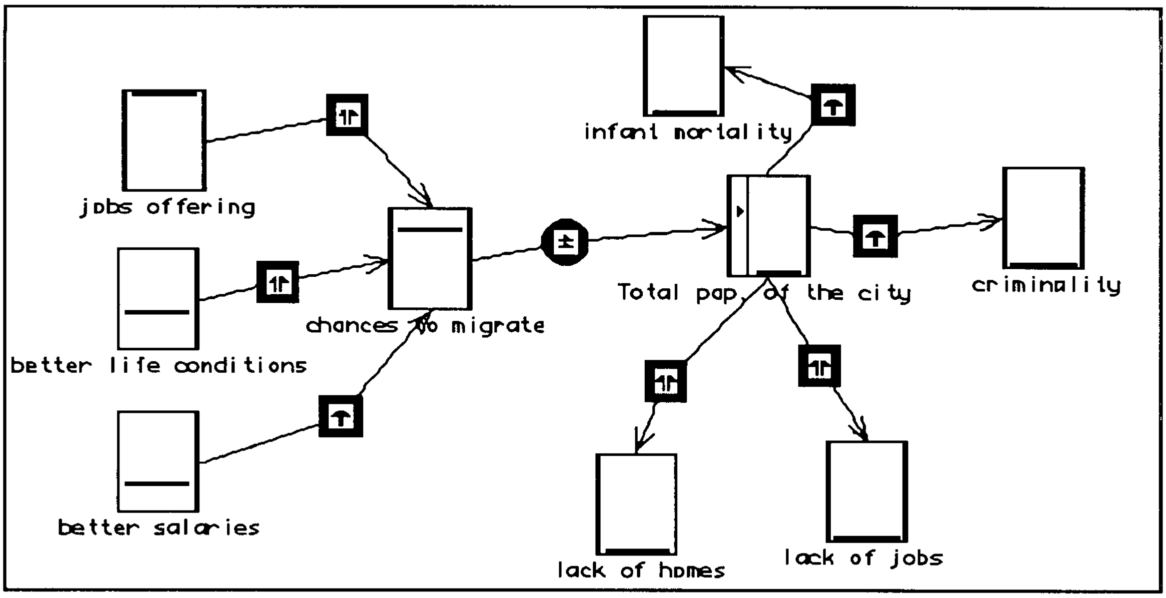


Figure D.6: First model about Migration to big cities

The use of *total population of the city* as 'On/Off' was to represent the idea of the bad factors beginning to happen after a certain level of population is reached.

They connected *chances of migration* to *total population of the city* via a 'cumulative' link because " (...) *if the chances are high population will increase every day* (...)". The idea was that every day the population would increase a bit, therefore it should be 'cumulative'.

After running the model they did not like to see the bad consequences of an over populated city staying at the same level of *total population of the city.* They then changed the link between them to 'weak'. The reason for doing that, seemed to be related to the interpretation they were giving to the levels. The same height means everyone. If, for instance, *lack of houses* ends in the same height of *total population* that means that not all the population have a place to live.

After that they decided to make their model more realistic, inserting a negative feedback loop to control the causes of making people migrate to the city. Again they used 'cumulative' links because they were seeing things changing "*gradually*".

At the end of the task I asked them if they thought the model was realistic and if it would be possible to use it to make some tests and they said:

> P: " *If someone wants to know what would be the chances* (to migrate)... *things like that... he knows jobs offering is very high... life conditions is small... better salaries is good... he wants to know the chances of migration... he is going to know there in this case... it is going to calculate*

the statistics of the population... by how much it will increase.. how long it will take to increase... and the problems that will happen..."

Task 5 - Learning a new subject matter - Activity 1 - Exploring a model about predator-prey

An interesting point here was that after reading the text and doing one simulation with the model (high level of *rabbits* and middle of *foxes*), they started saying that the model was correct and began to justify it using the explanation in the text as though they were seeing it happening as the model was running.

They more or less repeated a behaviour where one simulation was enough to answer a certain question. Only when I challenged them did they go on to do more tests.

At the end of the task I asked them how realistic the model was and they said it was correct because it was in accordance with what was in the text, also mentioning the idea of the feedback. Then I asked them if it was possible to "break this cycle" and they started talking about the human beings interfering in the process by cutting trees which would cause a lack of food for some animals. I asked them to implement it and they decided to create a new variable *food of the rabbits* connected to *rabbits*. Although they were seeing the relation between them as 'go together', it was created as 'cumulative' because of other 'cumulative' links arriving at *rabbits*. They ran the model and did not see *rabbits* being exterminated. They then decided to connect *rabbits* to *food of the rabbits* in order to make food diminish and, through the feedback, to make itself diminish. After running the model again and not seeing it happening Diego suggested making *food of the rabbits* 'any value' to represent the idea of "*eating less than necessary* (to survive)" (See FigureD.7).

Pedro did not agree with him becuase it did not make sense to have negative values for food of the rabbits. However, Diego convinced him to make the change and try another simulation. This time the model worked in the way they wanted. After that, they started agreeing that the "zero" point of *food of the rabbits* was at the very bottom of the box, the middle was "eating what is necessary" and above it was "eating more than enough".

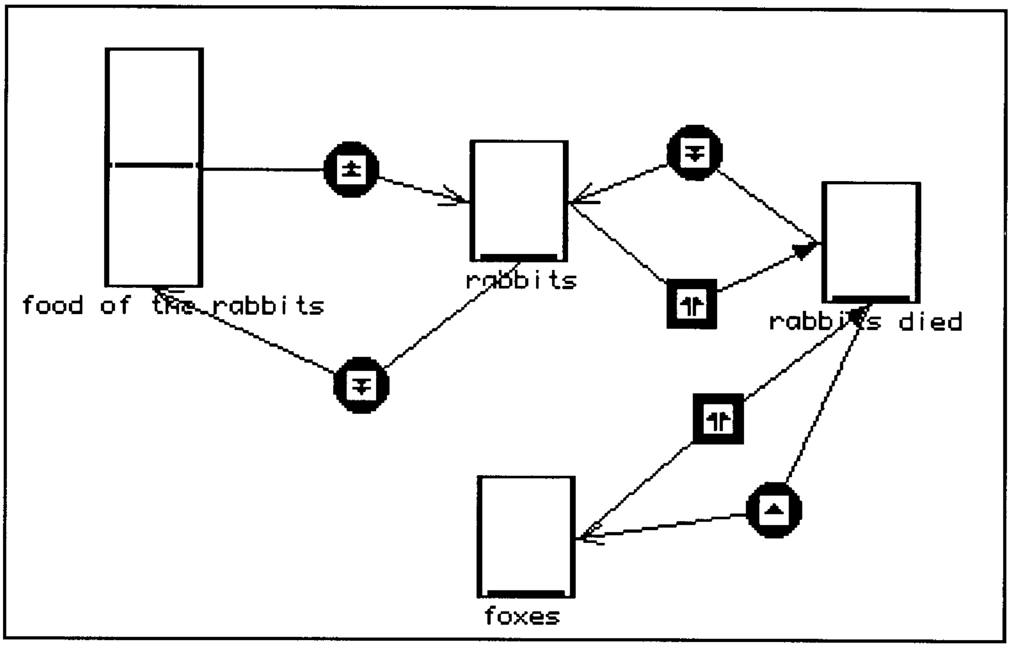


Figure D.7: Improved model about Predator-prey

Task 5 - Learning a new subject matter - Activity 2 - Exploring a model about eye-pupil

The model I showed them had the direction of two links changed from that in the original task (between *excess of light* and *size of pupil* and between *size of pupil* and *light in the eye*) in order to make the variable *size of pupil* behave in the wrong way (See Figure D.8).

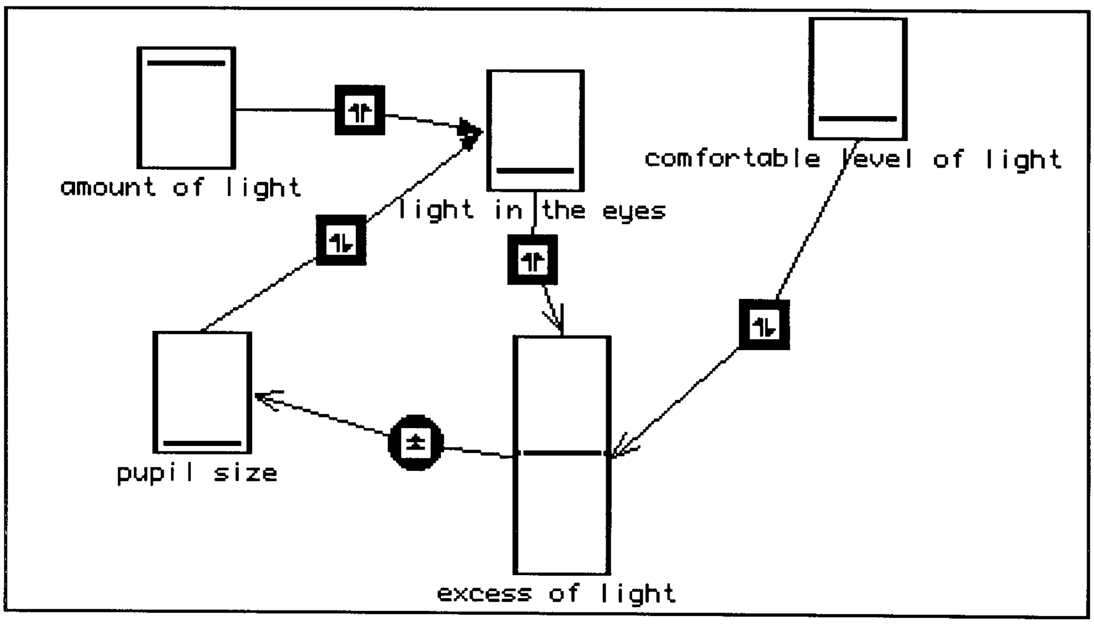


Figure D.8: Model about Eye-pupil presented to the students

When they first ran the model they only set the values of *amount of light* (very high) and *comfortable level of light* . Because they knew a bit about the problem in real life, they were expecting the *size of pupil* to go down (In the text I gave them there was nothing about what happen to the eye-pupil in specific situations such as in bright and dark places). However, it did not happen and they thought the model was incorrect.:

D: " *It is wrong...It should be the contrary, it should be low... If you have a lot of sun your pupil is going to be smaller... if you're in the darkness, it increases. When you're in a bright place it diminishes.*"

When they attempted to fix the model, Diego mentioned that the link between *pupil size* and *light in the eyes* was wrong and immediately after opening its Information box, he noticed that this link was being multiplied by the one that comes from *amount of light.* They changed the link to 'same direction' and set the combination to 'average'.

To test whether the *eye-pupil* opens or close when you go to the beach on a very sunny day, they just tried one simulation with a high value of *amount of light.* Only after I challenged them about "how was the situation before they go to the beach", they said it would be necessary to do two tests.

The interesting thing was that later, when they went on to demonstrate what happens to the eye-pupil when someone comes out of a long tunnel they spontaneously said that it would be necessary to do two tests:

P: " (...) *it would be necessary to do two tests, like the last time, inside the tunnel...now he gets out of the tunnel, the quantity of light is going to increase... is going to increase a lot ...and the eye pupil is going to be like this... it is going to diminish*"

Later, when they went on to describe the variables of the model, they began to discuss about the idea of blocking the eyes completely with an eye-patch and whether the model could be used to represent it or not. The interesting point was the fact that they brought in other ideas to be discussed and the validity of the model to represent them.

Task 5 - Learning a new subject matter - Activity 3 - Working with the model about refrigerator and heater

This was the first task of the third meeting. Only Diego was present.

I started asking him about his ideas on how a refrigerator works under different conditions (inside a house and outside on a very hot day). His idea was that in the sun the refrigerator would not work well because the sun would defrost the products. However, inside the house it would work properly, freezing the products in the "*normal way*".

After loading the model and running it for the first time, he was able to describe what was happening to the model talking about its variables and relations, also using information from the outside world:

> D: " *The external temperature is very high...when the sun is shining. But there is a great difference between the temperatures. The temperature inside the refrigerator became the same as the thermostat of the refrigerator... If the thermostat is high... the temperature goes down.*"

He also noticed the maximum value reached by the *temperature inside the house* could not go beyond the threshold of the *thermostat*.

After that I asked him whether he knew what a thermostat was and he said:

> D: " *Is it its thermometer ?* "
> F: " *Its thermometer...? and so what does it do with the refrigerator ?*"
> D: " *When it reaches a certain point... I don't know...When it reaches a certain point it cools the refrigerator more, then its* (refrigerator) *temperature falls*"

It is clear from the passage above that he did not know very much about the *thermostat* but after seeing it working in the model, he was able to construct a meaning for it.

After that he went on to answer the questions in the worksheet. He first ran the model with a lot of *external temperature* and noticed that *temperature in the refrigerator* dropped very much. He ran the model again with less *external temperature* and this time he did not like what he saw because *temperature in the refrigerator* ended at the same height as *external temperature*. The way I interpreted his behaviour was that he was still thinking that outside the house was "hot". After I told him that it was "cold", he began to see the values of the variables in a more relative way.

He also focused closely on the behaviour of *thermostat* under different conditions and began to realise the *refrigerator* works more on sunny days because the *thermostat* "*keeps turning on and off* (in order to make) *the temperature inside the refrigerator stay low*" .

When I asked him to measure the consumption of energy he correctly created a variable called *expenditure* connected to *thermostat* via a 'cumulative-same direction' link. After he explained the difference between using 'cumulative' and 'go together':

F: " *Why ball* ('cumulative') *?* "

D: " *Every time thermostat turns on, here* (expenditure) *will be increasing* "

F: " *If I want to measure for one month how much I spent. Would you use ball or square ?*"

D: " *Ball...*"

F: " *When square* ('go together') *could be used then ?*"

D: " *Only for once*" (meaning that it would serve to show just the first turning on of the refrigerator)

Model about the heating system

When I asked him about the construction of a heating system he told me he had no idea how it works but he could try.

During the construction of his model some interesting issues stand out .

For him the mechanism of the heater was controlled by a person. This variable, a bit later, was named *turn on*:

D: " (...) *the person turns on the heater, the heater is going up... indoor temperature as well...increasing...When indoor temperature reaches here* (the threshold), *the person turns off the heater, because it is OK inside the house, and here* (heater) *goes down as well*"

...

F: " *Then what is the heater ? It has a temperature* (sensor) *inside it, does it ?*"

D: " *No. this box* (variable representing the person) *here is going to change it... it would be the person that turned on the heater. When the person turns it on.... This relation here has to be square* (go together)... *it has to be the same* (...)"

The passage above suggests that he made the link between *turn on* and *heater* 'go together' because he wanted something that happens instantaneously (See Figure D.9).

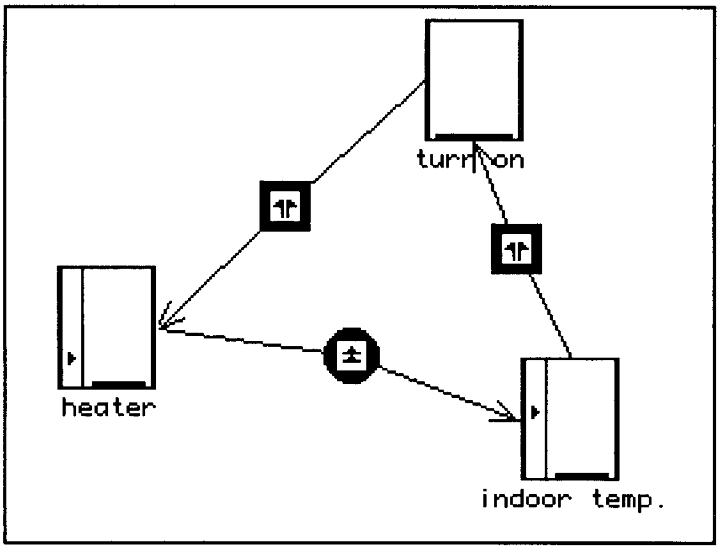


Figure D.9: Model about a heating system. *Heater* turns on when it is 'above the level' and *indoor temp.* turns on when it is 'below the level'

Only later, after discussing his ideas with him, his interpretation about the variables *heater* and *indoor temperature* became clear for me. *Heater* was 'On/Off' because it was to represent a situation where the heater (when it is on) became hot but was not enough to warm the house. Only when it reached the threshold level it began to raise the *indoor temperature* bit by bit. This idea also justifies the 'cumulative-same direction' link between them.

The threshold of *indoor temperature* was to represent the maximum value of the temperature inside the house. Above it the person has to turn off the heater.

Another interesting point here was that during our discussion he asked me whether it would be possible to make *indoor temperature* with two thresholds. The second threshold was to make the person turn on the heater below a certain value of *indoor temperature.*

While constructing this first model he was doing some simulations to see whether his ideas were working. When he completed the model shown in Figure D.9, it seemed he was satisfied with what he created using a explanation that only considered the system working under warm temperatures. I then decided to challenge him asking him to imagine a cold winter. He then decided to introduce a new variable in the model called *cold* and connected it to *indoor temperature* via a 'cumulative-opposite direction' link.

Task 6 - Express their ideas about a complex problem - Diet and healthy life

After reading the text Diego decided to represent in his model the main ideas about diet and health. He first created two variables *diet* and *health* and connected them via a 'go together-same direction' link. After doing a test, he decided to insert a new variable which he named *quality of the food* and later when he decided to introduce the four main components of the diet (carbohidrates, proteins, fat and fibres), he changed *diet* to *right diet* (meaning good or healthy diet. See Figure D.10)

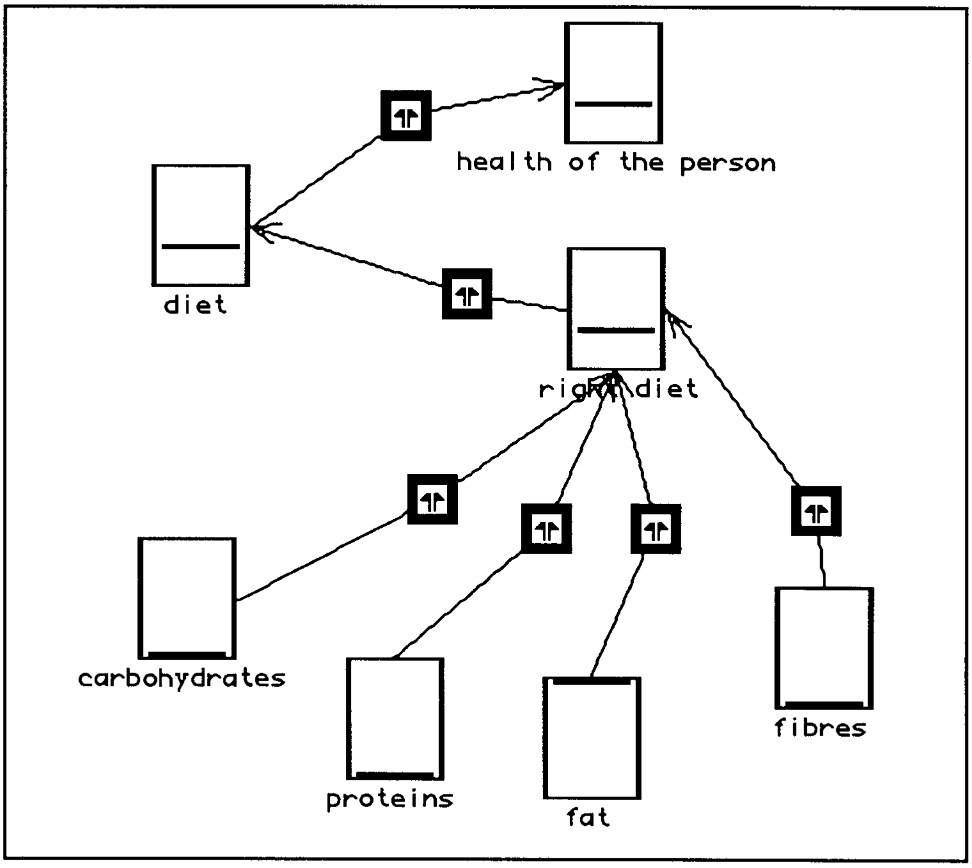


Figure D.10: First model about Health

He decided to use 'average' to combine the main components of the diet. The way he justified it was that the final value (*right diet*) should be somewhere in the middle of.its causes. While doing it he made a comment that if he changed the 'input combination' in one link all the others would be changed as well. The way he said it indicated that he was asking for my confirmation of what he was doing. This makes me think that so far he was not sure about which things are combined when you change this parameter.

Another interesting point was that although he was seeing *fat* as a negative cause for a good diet he did not change its outgoing link to 'opposite direction'. He made it 'weak' instead saying that "*the correct ithing is not to eat too much fat*".

After being satisifed with his model, I decided to ask him what happens when health becomes bad. He promptly suggested a new improvement in the model making *health* as 'On/Off' and creating a feedback mechanism to improve the *quality of the diet* (See Figure D.11).

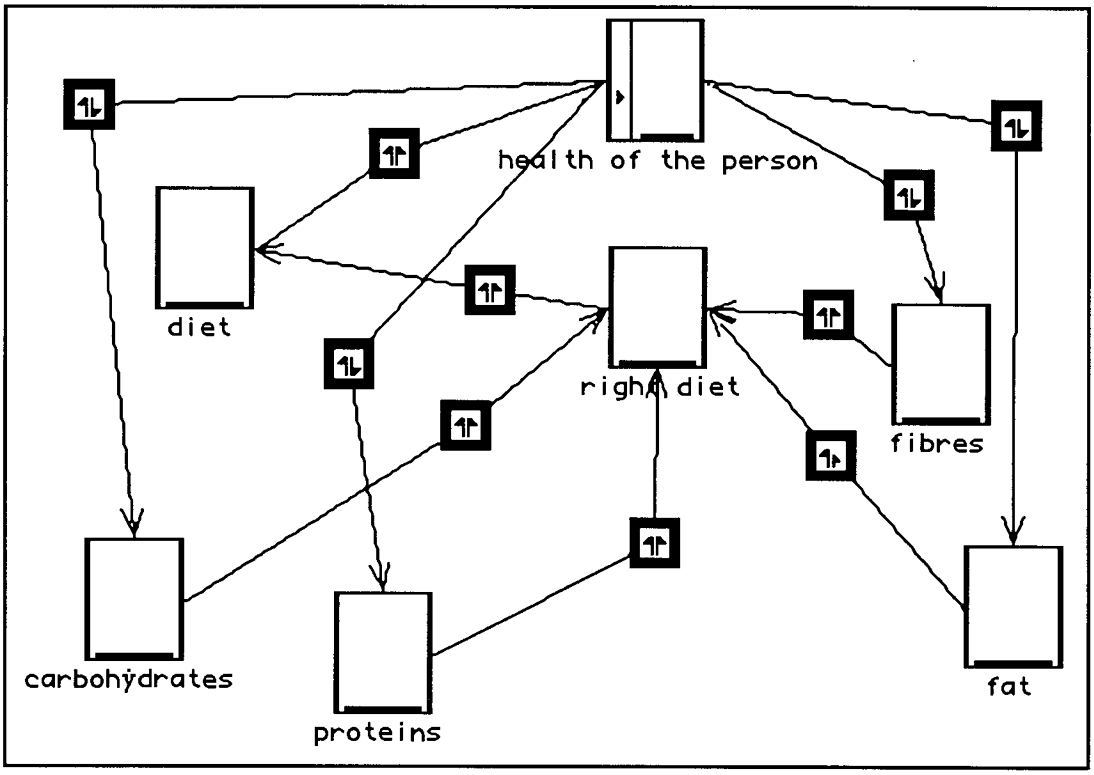


Figure D.11: Later model about Health

To justify the 'opposite direction' links between *health* and the main components of the diet he used an argument that clearly demonstrates that he was seeing them as an inverse relationship:

> D: " (...) *when his health is bad he has to eat more fibre, more fat, more proteins and more carbohydrates...*"

A bit later, after running the model and not seeing what he was excpecting he also said:

> D: " (...) *but it has to be opposite because the lower the health the more he has to eat that stuff* "

I was convinced that he was seeing the relations as inverse and decided to make him focus his attention on those links. After giving an explanation about the idea of 'opposite' and 'inverse' using absolute numbers, he changed the links to 'same direction' and ran the model again. This time the simulation showed what he wanted to see and he was satisifed.

It was clear to me from our discussion that he only used 'go together' links because he was seeing the model as an event that just happened once. The model was there to demonstrate that if someone has a bad diet, his health will be bad and he has to improve the food he eats. In the passage below he was giving me an example to justify why the model constructed served to represent the problem in real life (this was a question I put to him):

> D: " *If he is a very poor person.... when he begins to eat only fat... you leave only fat, proteins, carbohydrates and fibres low... the diet will not be correct. It will be low. Diet will go down... the health of the person will go down... (and) when it goes down, he has to find a way...to eat... in this case I don't know* (he probably doesn't know how a poor person will have money to buy proper food) *...to make health become good again*"

Task 7 - Discussion about empty models

This task was presented at the beginning of the fourth meeting. Both Diego and Pedro were present.

I showed them three different empty models (all variables lack names) using different properties of the system.

In all cases, before they went on to suggest a situation that fitted to the model, they ran the model with some initial values and inspected the properties set to the variables.

First empty model

The first model had the general form:

XXX -- **Cum,same** -->
------ add --> ZZZ
YYY -- **Cum, oppos** -->

The kind of model I had in mind was

Rate of birth -- **Cum,same** -->
------ add --> Population
Mortality -- **Cum, oppos** -->

They suggested to me the following model

Deposits -- **Cum,same** ---->
------ add --> Account in the bank
Withdraw -- **Cum, oppos** -->

During the discussion we had they changed their interpretation of the variables. Sometimes they saw*deposits* as the number of deposits and sometimes as the amount deposited.

They justified the links as 'cumulative' because they saw the problem as money being accumulated over a certain period of time:

> F: " *If the number of deposits is fixed... lets say 100 the account is not going to increase and increase... it is going to stay in a fixed level...*"
> D: " *What I'm saying is: I'm not saying 100... I'm not talking about exact numbers... you make a big number of deposits per month, your account in the bank is increasing* (meaning bit by bit)..."

Later, after I accepted their model, I asked them how they would represent a situation where someone makes a fixed number of deposits in one month and I said I wanted to see the situation only for one month. They said that in this case the link should be 'go together'.

Second model

The second model had the form:

XXX -- **Cum,same** ----> YYY
/\ ------**Cum, oppos**---------\/

The model I had in mind was:

Food avaiable -- **Cum,same** ----> Population
/\ -----------**Cum, oppos**------------\/

The model they suggested was:

Money you have -- **Cum,same** ----> Shopping
/\ --------------**Cum, oppos**---------------\/

Their idea was about someone having a certain amount of money and going shopping. When he/she begins to buy things his/her money begins to go down.

The idea sounded good but when they ran the model they expected *shopping* to go down to zero after *money* finished. I then decided to let them "fix" the model in order to make it behave in the way they wanted. They first thought about making *money* you have as an 'On/Off' which would turn on when the level goes below the threshold. When it turned on it would make *shopping* go to zero (See Figure D.12)

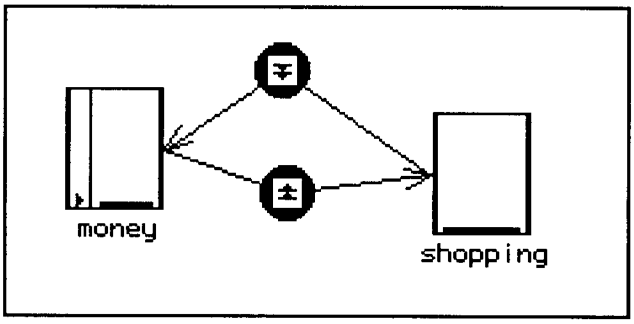


Figure D.12: Model about Money and shopping

But when they ran the model with level of *money you have* above the threshold, they did not noticed any change in the variables. The real reason for this was the use of 'cumulative' links and the fact that when *money you have* was off it makes its output equal to zero. However, they decided to make *money you have* 'smooth' again and *shopping* 'On/Off' triggering when above the threshold (Here they simply transferred the problem in their model from one variable to another). Again the model did not work. I then decided to make them think deeper about the way they were using the On/Off variables and in the end they realised that they should change both links between *money you have* and *shopping* to 'go together'. They ran the model again and were satisfied saying:

D: " *It is correct...*"

P: "*Because when...*"

D: " *Because when he has money he buys. When money finishes he stop buying.*"

An important point which was behind their ideas and which they did not consider was about what they wanted to represent with their model. They were moving between the idea of having money and going shopping (talking about an event that happens or does not happen) and the idea that while someone has money he/she goes shopping and buys different things (talking about accumulating things). This is a very important differentiation that should be made in advance. However, this is a big leap in the process of modelling which even experienced researchers do not always make. In a real school activity the teacher, together with the system, could lead them to think further about these different ideas.

Third Model

The third model had the form:

XXX -- **Cum,oppos** ----> YYY ---> **Immed, same** --> ZZZ (On/Off)
/\ -----**Cum, same** ------------\/

The model I had in mind was:

Winter - **Cum,oppos** -> House Temp. -> **Immed, same** -> Heater (On/Off)
/\ --------- **Cum, same** -----------------\/

The model they suggested was:

Hot - **Cum, oppos** ---> Cold -> **Immed, same** -> Air Conditioning (On/Off)
/\ ------------ **Cum, same** ------------\/

Before they suggested a problem that fitted the model , they first ran the model and started "reading" it as they were running it manually. The interesting point was that they only considered the variables increasing:

> D: " *Bigger the first* (variable)... *smaller the second. Bigger the second... it is going to be the same as the third.... Bigger the third... it will be bigger with weak effect on the other* (first variable). "

But when they went on to suggest a problem, they "read" it in another way:

> D: " (...) *bigger the hot* (first variable) *smaller the cold* (second variable).... *Smaller the cold... smaller the heater* (smaller the need to turn on the heater)..."

At this time they "read" exactly what was on the screen.

After running the model they realised that the suggestion did not fit the model well because the 'On/Off' variable was triggering when 'below the threshold'. Then they immediately suggested changing the *heater* to *air-conditioning.*

Because the model was not running well I left them to change the weight of the link between *air-conditioning* and *hot* to 'strong'. Before they did it Diego correctly said that it was not working because the value that was passing from *air-conditioning* to *cold* was zero (The trigger level of the variable was very low. Therefore when it turned on its output was almost zero).

Model proposed by them

The way they constrcted their model was interesting. They were thinking about a bank functioning. They first considered the *deposits*, *withdrawals* and robbery. They created the model (See Figure D.13) and began testing it with different initial conditions.

They did not explicitly justify the use of 'cumulative' links but it was clear from their dialogue that they were thinking about the events happening during a certain period of time.

Another interesting point was the fact that they combined the links using 'average'. Again from their dialogue it seemed that the choice of 'average' has to do with adding and subtracting values and achieving a result that has to be in the middle of its causal factors. Later when I had to find a problem that fitted their model, I suggested that the variables going to *money in the bank* should be 'add' and not 'average' and they said that "*it also could be add*".

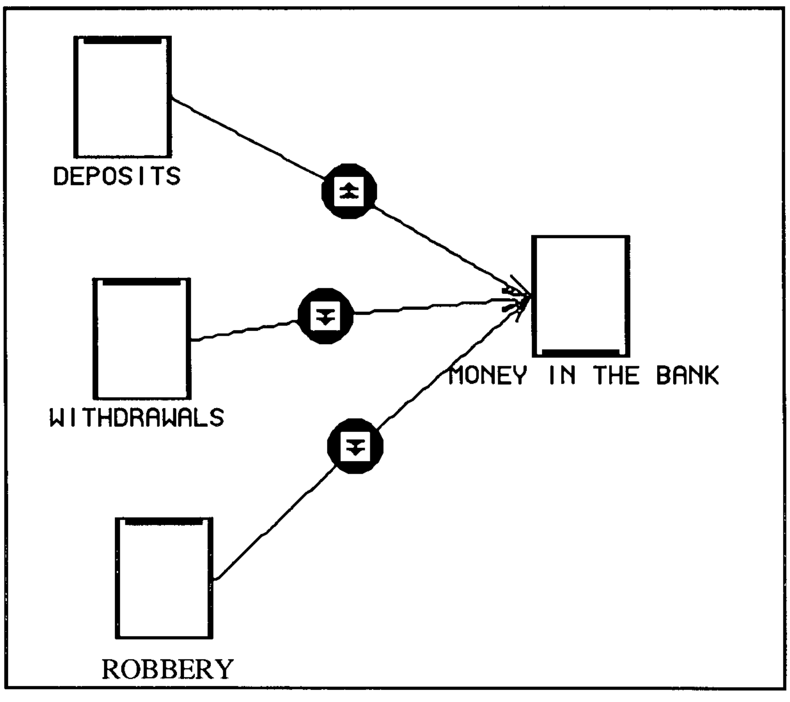


Figure D.13: Model proposed by the students about a bank working

When they were satisified with the simulations, they saved it and began to make the model more sophisticated. This time they focused on the *number of customers* of the bank and what causes its increase: *how well known is the bank* and *how good is the service* (See Figure D.14).

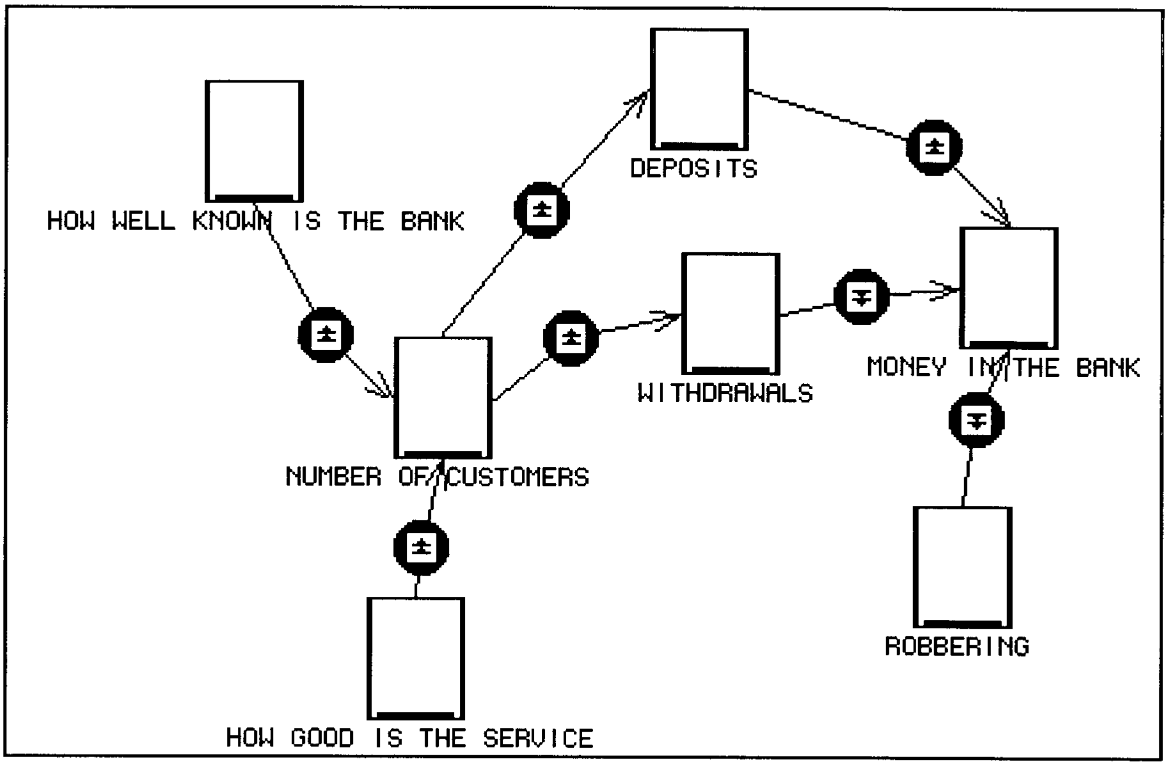


Figure D.14: Improved model about a bank working

It became clear during the discussion about which variable would connect to which that they were separating the problem about the bank into two parts: one related to the customers and another related to the money there.

D: " (..) *money in the bank doesn't have anything to do with number of customers...*"
P: " *Services... number of customers is going to influence... the money in the bank isn't it ?*"
D: *No...*"
P: " *Oh Yes. It is already* (influencing) *in the deposits* (...)"

During the elaboration of their model it seemed that the importance of *robbery* faded out (At least the word was not mentioned any more). These events made me think that they sometimes began a problem with an idea in their mind and when they came to the system, other issues began to evolve and have more importance. They then (more or less) abandoned their original ideas going on to more general and complex ideas.

Task 8 - Constructing a model about pollution in big cities

The model they decided to discuss was about the causes of the hole in the ozone layer. They first considered its main cause - *pollution of the air* - . which was caused by other factors. After creating these variable, they considered some factors (*filters and trees*) that would diminish pollution.

They decided to make all links 'cumulative' because they were not " *just thinking about one month* " meaning that they wanted to observe the over a long period.

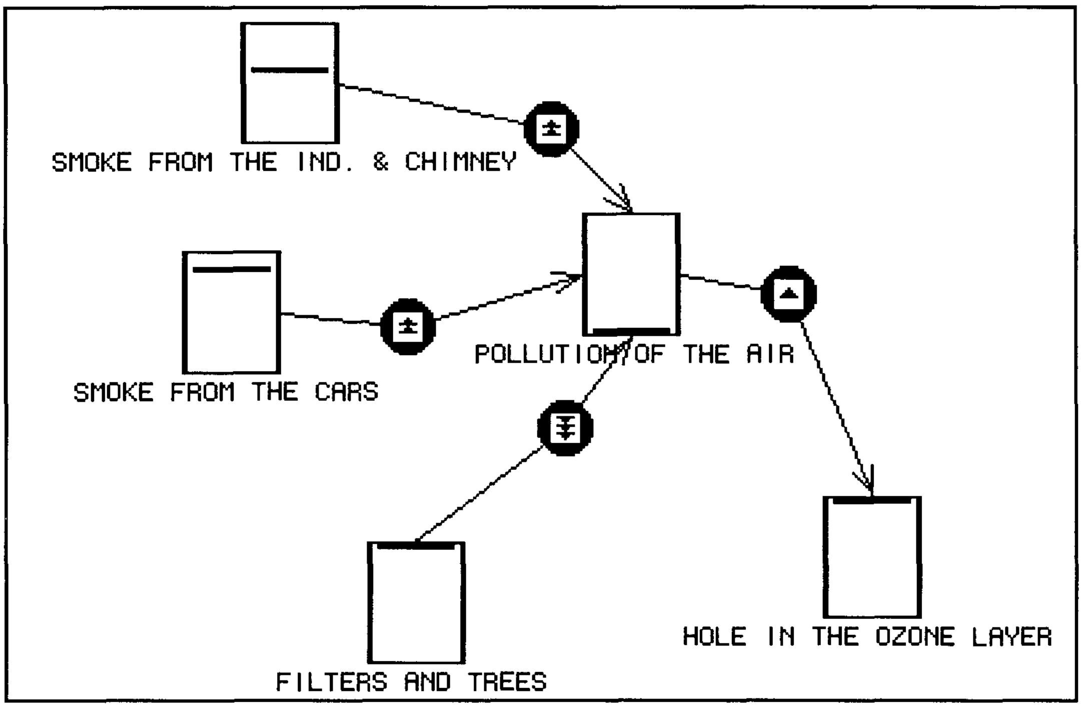


Figure D.15: First model about pollution of the air

After constructing the first model (SeeFigure D.15) we began to discuss what makes people *plant trees* and put *filters* in industries. They then began to develop a new idea about the government being responsible for this task. While they were talking about it they thought that *pollution of the air* could be 'On/Off' to represent the idea of *protests* beginning to happen only after a certain level of *pollution.* The same kind of argument was used to make *attitude of the government* 'On/Off': the government only takes up an attitude after a certain level of *protests* (See Figure D.16).

After trying some simulations they realised that some 'weights' were not very much appropriate because the effects on some variables (*pollution of the air* and *protests*) were happening too fast.They then changed the links between *filters and trees* and *pollution of the air* and between *pollution of the air* and *protests* to normal.

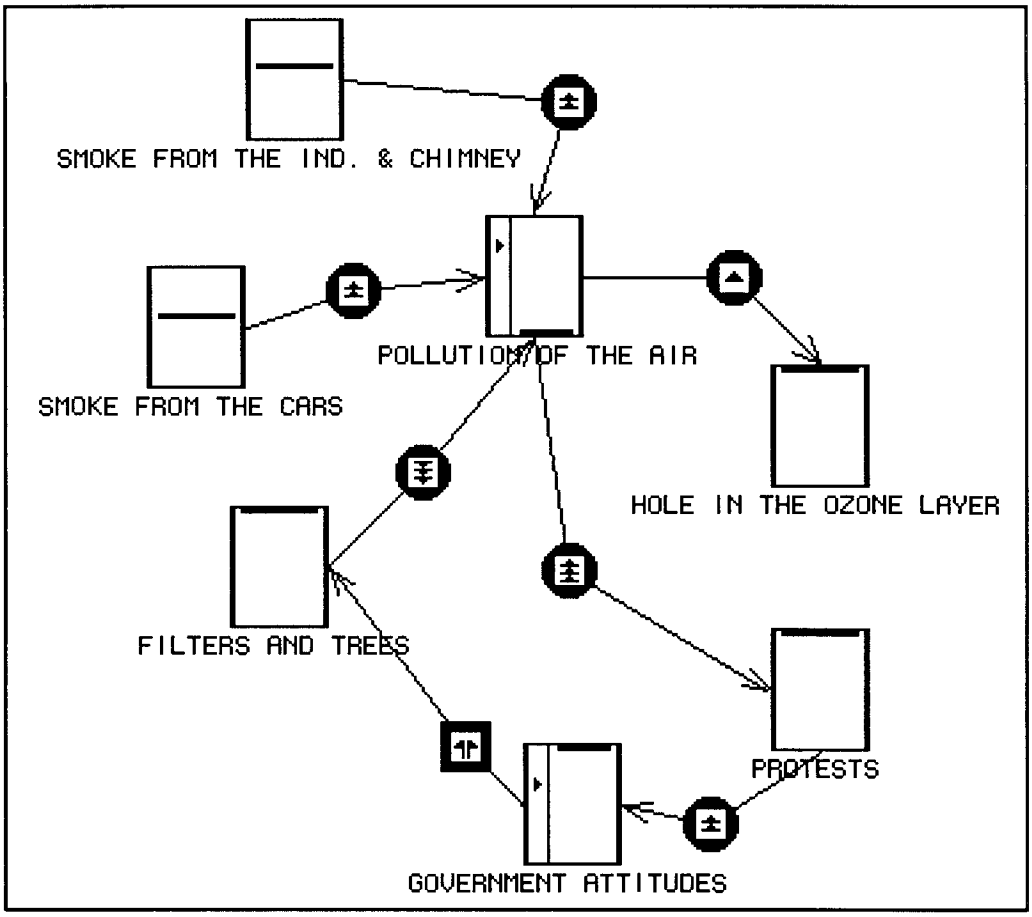


Figure D.16: Improved model about pollution of the air

After changing the model, trying some simulations and being satisfied with what they saw, I asked them to explain the model to me and they said:

> P: " *Here... smoke from the industries and cars... contribute to the pollution of the air. Pollution of the air is going to contribute to the hole in the ozone layer. When pollution arrives at a certain level... the people will be angry...* (and) *they will start to protest ... then the government is going to begin to work... to act. When it arrives at this point here* (the trigger level of government)... *it begins to construct filters and plant trees... which in its turn makes pollution diminish.* "

Their description suggests both a general understanding about the problem and how it is represented by the model.