METADATA-SUPPORTED AUTOMATED ECOLOGICAL MODELLING

Virgínia Brilhante* and **Dave Robertson**

School of Artificial Intelligence - Division of Informatics - University of Edinburgh
80 South Bridge
Edinburgh EH1 1HN
{virginia,dr}@dai.ed.ac.uk
*On leave from University of Amazonas, Brazil.

1. Introduction

Ecological models should be rooted in data derived from observation, allowing methodical model construction and clear accounts of model results with respect to the data. Unfortunately, many models are retrospectively fitted to data because in practice it is difficult to bridge the gap between concrete data and abstract models. Our research is on automated methods to support bridging this gap. The approach proposed consists of raising the data level of abstraction via an ecological metadata ontology and from that, through logic-based knowledge representation and inference, automatically to generate prototypical partial models to be further improved by the modeller.

In this chapter we aim to: *1)* give an overview of current automated modelling approaches applied to ecology, and relate them to our metadata-based approach under investigation; and *2)* explain and demonstrate how it is realized using logic-based formalisms.

We give the overview of current automated modelling approaches in section 2, focusing on Compositional Modelling and Model Induction. The contrast between these and our approach, where we adopt metadata descriptions through an ontology and logic-based modelling, is discussed in section 3. Section 4 makes ideas more concrete, starting with further details on the *Ecolingua* ontology (section 4.1), followed by examples of automated model structuring (section 4.2) and parameter estimation (sections 4.3 and 4.4). We conclude with section 5 commenting briefly on the ontologies trend and on the outlook of our research.

2. Ecological Modelling and Automation: Current Approaches

Ecological models are a widely used and powerful tool in environmental studies. Through the design of such models, modellers are able to give shape to their understanding of ecological systems and thus discover possible knowledge gaps. Once built, models can be used for ecological simulation, giving support to prediction and decision making. Initiatives exist to make accessible the large body of knowledge generated by ecological modelling efforts worldwide. The Register of Ecological Models, REM, is a comprehensive repository of ecological models, equipped with search facilities, available at http://dino.wiz.uni-kassel.de/ecobas.html ([Benz and Voigt, 1996], [Benz and Knorrenschild, 1997]). The models in REM are documented under a unifying documentation system called ECOBAS [Hoch et al., 1998].

Automating the construction of models in general is the ambition of the **Automated Modelling** field. Modelling processes of various disciplines are investigated and the modelling knowledge is explicitly represented, aiming at developing computer tools able to automatically follow modelling principles [Xia and Smith, 1996]. **Compositional Modelling**, the dominant automated modelling approach, consists of assembling basic model fragments, usually gathered in a library, into an well-defined model. In [Rickel and Porter, 1997] we find an application of compositional modelling to an ecology-related domain, namely plant physiology, chosen to demonstrate automated modelling of complex systems. The domain knowledge used by TRIPEL, the automated modelling system, is part of a large multipurpose biology knowledge base. The core of the system is an algorithm that given a prediction (what if) question, the variables of the physical system, the influences among them, and other domain knowledge, outputs the simplest differential equation that can adequately answer the prediction question.

In the **Model Induction** approach, investigated within the Machine Learning field, models are derived from data. Figure 1 summarizes the Machine Learning task viewed from a modelling perspective. Data, together with some background knowledge, is used to provide examples from which a general model structure can be induced (learnt) and described in some language such as if-then rules, equations, decision and regression trees.

One important motivation for this form of knowledge induction is that it is a promising alternative for ill-defined or poorly understood problems that lack algorithmic solutions [Kubat et al., 1997], characteristic of the environmental sciences. In other words, given that we fail to formally state the problem and all we have are factual examples, machine learning attempts to extract concepts that are embedded in the examples.

The series [Kompare et al., 1994] and [Dzeroski et al., 1997] describes the use of machine learning techniques on automated modelling of ecological systems. The first paper discusses early results of machine learning systems running over ecological data sets. The system RETIS [Karalic, 1992] induces regression trees from the Lagoon of Venice data. LAGRANGE, an equation discovery system ([Dzeroski and Todorovski, 1993], [Dzeroski and Todorovski, 1995]), has also been applied over the Lagoon of Venice data. Although the system performs well on synthetic data, it struggles with insufficiency, inadequacy and inaccuracy in the Lagoon data, typical problems in ecological databases. Other systems (GOLDHORN [Krizman et al., 1995], LAGRAMGE [Todorovski and Dzeroski, 1997]) succeeded and greatly improved LAGRANGE, but the problem of generation of many spurious or hard-to-interpret equations remains. Besides, these systems can only handle a limited range of data sets. A recent survey of this family of equation discovery system applied to ecology is found in [Dzeroski et al., 1999].

The second paper in the series presents the use of rule induction algorithms to classify water quality of British and Slovenian rivers, sampled with respect to biological and chemical attributes. The induced models are if-then rules, a different form of model description from the traditional differential equations models. Given examples of classified samples, general rules are induced that capture the classification knowledge, and can be applied to classify new samples.

3. Our Automated Ecological Modelling Avenue

Models and Data

The Compositional Modelling approach, to date, does not take into account the intrinsic relationship between models and data, amply recognized in the empirical sciences:

> "Science consists of confronting different descriptions of how the world works with data, using the data to arbitrate between the different descriptions, and using the best description to make additional predictions or decisions. These descriptions of how the world might work are hypotheses, and often they can be translated into quantitative predictions via models." [Hilborn and Mangel, 1997]

The common real-world practice, in Ecology and other domains, is to have a database generated via field experiments together with literature data as one of the pillars (human expertise is the other) on which models are built and calibrated.

By contrast, the TRIPEL system for instance, takes as modelling building blocks pre-determined influences among the physical system variables. The complete picture of all modelled variables and influences in the system are known in advance. The compositional modelling algorithm then searches for the smallest portion of this complete picture that is able to answer the prediction question. The possibility of having data revealing unsuspected influences between variables is not investigated.

Model induction, on the other hand, does rely on data and it is certainly a seductive idea. Algorithms outperform human data analysts on the task of extracting patterns from data. However, state-of-the-art model induction does not cope yet with real-world scale databases and models. We believe it is unlikely that model induction will fully succeed in complex domains like Ecology, due to the fact that data in these domains is always partial. Actual databases are larger (which by no means make them more complete), more complex and more noisy than the usual single-tabled databases tackled in published experiments. Moreover, problems of real interest demand models of a level of complexity that cannot currently be delivered.

Models and Metadata

In our research we aim at using data to support partially automated construction of ecological models. System dynamics, in particular, is the modelling paradigm we are experimenting with ([Haefner, 1996], [Grant et al., 1997], [Ford, 1999]). Our usage of data, however, is rather different compared to model induction. Instead of data values, we exploit data ontologies, descriptions of properties of data.

The intuitive motivation behind this idea is the way that model designers connect data to models. Only in later stages of the modelling process are data values directly manipulated, often for estimation of individual parameters. During early design stages however, a more subjective kind of data analysis takes place, where modellers ponder if the data at hand supports their understanding of the ecological system, which will ultimately be represented by the model. In doing that, the analysis focuses not on data values as such, but rather on higher-level properties of the data, or metadata. That is, availability, meaning and types of measurements, how the data is structured (e.g. organizational levels such as the grouping of variables associated to different ecological entities), how representative the samples are with respect to spatial and temporal requirements and coordination (e.g. sampling campaigns and frequencies, spatial distribution of sampling plots and points in each campaign). In short, metadata is the abstraction that bridges data values and domain knowledge.

We find support to our position on formalizing metadata and strengthening its role on automated modelling in the Intelligent Data Analysis field:

> "Formalizing metadata, in particular, formalizing it to such an extent that it, like data, can be manipulated automatically by computer, is one of the major challenges of modern intelligent data analysis." [Hand, 1997]

Figure 2 compares traditional modelling with the metadata-supported modelling approach. In the former, the human "eye" looks at an ecological system and, in the light of its perspective of the system, selects data items of interest. The model is designed based on the modeller's view of the system, and on a meta-level interpretation of the data (as we argued in section 3). In the latter, the starting point is to produce (manually) a meta-level description of the data (or metadata), to be combined with representations of modelling knowledge. Descriptions on hand, heuristics are applied to hypothesize partial models. The engineering of these heuristics is crucial; the connections drawn, in traditional modelling, between data and models ought to be unraveled and reflected in them. Lastly, equipped with these partial models which were automatically derived, the human modeller can improve them according to his or her understanding of the ecological system to construct a complete model which is more clearly connected to the data.

An Ecological Metadata Ontology

An automated modelling system based on metadata can only be attractive if it is able to handle descriptions of diverse data sets. Predefining a detailed knowledge representation system general enough to express meta-level properties of every ecological data set is infeasible. Thus, what is needed is a general unifying framework that can be instantiated to specific databases. This brings about the benefit of enabling the handling of compact descriptions of data rather than large volumes of data values.

A framework of this kind is an ontology: a shared understanding of some domain of interest, specified in the form of definitions of representational vocabulary and axioms that constraint interpretations over this vocabulary [Uschold and Gruninger, 1996]. We have been developing an ontology, named *Ecolingua*, for expression of ecological metadata (section 4.1). The collection of concepts is structured as a hierarchy of interrelated classes, in a style similar to frame (or object-oriented) systems. This is in line with previous work on structuring and formalization of ecological knowledge, such as [Lorenz et al., 1989] and [Uhrmacher, 1995], with the distinction that our classification applies not to model structure but to the properties of data which should influence our choice of model structure.

Logic-based Ecological Modelling

The ability to infer new information from the metadata descriptions is crucial to our project, since we aim at finding connections between metadata and models. This can only be achieved if we can transform, through inferences, the two forms of knowledge - metadata and models - into each other. When doing this we are often concerned to know when the meaning of statements are equivalent, which is a motivation for using logic.

Computational logics are now highly developed and it is relatively easy to build standard forms of inference procedure. Non-standard inference mechanisms dealing with rationale and argumentation are also beginning to appear and these can be used to make known to the system user how the inferred models are endorsed by the metadata.

The Eco-logic approach, compiled in [Robertson et al., 1991], proposes logic programming as an alternative for the construction of ecological simulation programs. Emphasis is placed on the use of domain knowledge to support modelling automation, making model assumptions explicit to enable more informed model simulation analysis. Our work evolves from these ideas, adding to them by investigating how ecological metadata (which play a part in domain knowledge) can be conducive to model construction.

Early work in this line is [Muetzelfeldt et al., 1989], where system dynamics ecological models are represented and simulated in Prolog, a logic-based programming language ([Bratko, 1990], [Sterling and Shapiro, 1994]). It is advocated that the use of Prolog improves both the rigor of the modelling process and accessibility of modelling to ecologists. Models, rather than being black-boxes, have their complete structure made explicit and transparent.

[Uschold et al., 1984] presents ECO, an intelligent front end for ecological system dynamics modelling. It allows a modeller to interact with the system in a free-form dialogue using ecological terminology. The system then integrates the modeller's input and a knowledge base of ecological data and ecological modelling information into a model, translates this model into a FORTRAN program and runs it. Following up, [Uschold, 1991] claims (and we agree) that there is a large conceptual distance between available modelling tools and the way users think about their problems. He proposes the Elklogic language, based on typed lambda calculus, which is suitable for representing both domain and simulation modelling information. The representation requirements for Elklogic come from an ecological modelling knowledge ontology, characterizing concepts such as *ecological entity* and *ecological attribute*, also present in our *Ecolingua* ontology. Elklogic also allows for attributes induction. Higher order functions are used to represent inferences such as 'if there is the attribute *weight* of a certain animal, then the attributes *average/total/maximum/minimum weight* apply to groups of the animal' or 'if there are several daily readings of temperature then the attribute *daily average maximum temperature over the course of a week* applies'. This feature of the language sheds light on our metadata↔model connections. Similarly, in section 4.4 we show an example where the estimation of the parameter *average annual increment of DBH of trees* is inferred from ecological metadata.

The outcomes of our project can contribute to the easing of problems raised by [Uschold, 1991]. He claims that his system offers, mostly, support in identifying and pruning the modelling search space. However, the knowledge involved in making design decisions is not available and the provision of heuristic guidance for these decisions is minimal. Our hypothesis is that part of this knowledge is embedded in ecological data and that it can be captured by meta-level descriptions of the data. Then, based on human modelling reasoning and practices, we can design and apply heuristics over the metadata descriptions to populate the search space with modelling elements that can be justified in the properties of the data behind the model.

4. Towards a System for Metadata-Supported Automated Modelling

Figure 3 depicts the envisaged complete picture of the **system architecture** . The starting point is a given ecological data set. One core trait of the approach is that we do not handle data values. Rather, the modelling reasoning is performed based on higher level, more compact descriptions of the data. In order to draw these descriptions some language is needed, through which data properties can be expressed. *Ecolingua* is an ontology built for this purpose (section 4.1). The original data set will not yet be suitable for modelling. Data set quantities in raw form usually do not directly map into or relate to model elements (also the case in human-centred modelling). A process, possibly iterative, of data composition is needed until some data-to-model mapping becomes possible. The composed data set is then described through *Ecolingua* language. With a described modelling data set on hand, we are ready to put the automated modelling engine into action. The descriptions of the data properties are the main input to the engine, along with supporting (also formally described) model requirements and assumptions, ecological and modelling knowledge, and user input. We distinguish three stages of the system dynamics modelling life cycle: *model structuring*, where the model's conceptual structure and functional relationships among elements are designed, *equation design*, where the model is mathematically defined, and *parameter estimation*, where values for the constants that calibrate the model are determined.

In sections 4.1 to 4.4 we present some results produced towards the realization of this architecture. The architecture components concerned are marked * in the diagram, followed by the section where they are discussed.

4.1. *Ecolingua*: a Formal Ontology for Ecological Metadata Description

Ecolingua is an ecological metadata ontology. It provides a vocabulary for formal description of ecological data properties. Upon these descriptions the automated modelling mechanisms draw modelling elements that are consistent with the data properties. We make no claim that it is the definitive ontology for this purpose - on the contrary we view it as a statement of our view of the problem, which others may challenge.

Figure 4 gives a conceptual graph of some of the *Ecolingua* classes and relations involved in the examples throughout this chapter, together with an excerpt of *Ecolingua*'s type hierarchy. Boxes represent classes, and ellipses, relations between classes. The classes constrain the type of objects involved in the relations. The directed arcs show the direction of the relation, from domain classes to range classes; e.g. the relation *sampling frequency* gives a *non-negative integer* (range class) as the number of occasions in which a *quantity* (domain class) has been sampled along a *year* (domain class). The type hierarchy represents class/subclass relations; e.g. *campaign* is a subclass of *time interval,* which is a subclass of *temporal thing*, and so on. The classes *thing*, *physical attribute*, *area*, *temporal thing* and *time interval* (dashed boxes in the figure) belong to the Hpkb-Upper-Level ontology, one of the ontologies available through the Ontolingua server from which we reuse definitions (Knowledge Systems Laboratory, Stanford University, http://www-ksl-svc.stanford.edu, [Farquhar et al., 1996]).

The classes and relations are meant to embody concepts that one would evoke in order to convey descriptions of ecological data properties. They function as a framework that is instantiated to the particular supportive database of the model. The instantiation is not done automatically. Thus, human judgement determines the classification of objects in the database and the establishment of relations between them. Certain classes are more permissive than others. For example, the ontology is not able to impose strong constraints as to what can or cannot be classified as an *entity. Entity* is an extensive concept that applies to anything that exists and is identifiable in the ecological system. Other concepts require more strict constraints defined as ontological axioms, such as sentences 4.1 to 4.4 below, specifying constraints for classification of quantities as being of types *amount* and *material rate* (referred to in subsequent sections). Axiom 4.1, for instance, specifies that *A being a quantity of type amount_of_material of M in E given in U, implies that M is a material, E is an entity, M and E are compatible and U is an appropriate unit of measure for amounts of material.*

$$\begin{aligned} amount\_of\_material(A, M, E, U) \rightarrow\ & material(M) \wedge \\ & entity(E) \wedge \\ & compatible(M, E) \wedge \\ & amount\_of\_material\_unit(U) \end{aligned} \tag{4.1}$$

$$\begin{aligned} abs\_mat\_rate(R, M, Eo, Ed, U) \rightarrow\ & mat\_rate(M, Eo, Ed) \wedge \\ & abs\_mat\_rate\_unit(U) \end{aligned} \tag{4.2}$$

$$spf\_mat\_rate(R, M, Eo, Ed, U) \rightarrow mat\_rate(M, Eo, Ed) \wedge spf\_mat\_rate\_unit(U) \quad (4.3)$$

$$mat\_rate(M, Eo, Ed) \rightarrow material(M) \wedge entity(Eo) \wedge entity(Ed) \wedge mat\_origin(Eo) \wedge mat\_dest(Ed) \quad (4.4)$$

A brief word on logic

This is the first time in the chapter we introduce a piece of logic. It may be convenient to explain the meaning of formal notation we use to the reader less acquainted with logic. If you want to find out more, two of the many introductory texts on logic are [Guttenplan, 1986] and [Lemmon, 1987].

Logical sentences describe arguments about the/a world. A sentence can be either a fact, a given statement (e.g. 4.6 in the sequel), or a rule (e.g. 4.1 above), that can be seen as a generalization of facts. The atomic components of sentences are predicates. They express properties of or relations between objects. For instance, the predicate *compatible(M, E)* represents a compatibility relation between *M* and *E,* being *M* and *E* variables that can become instantiated to constants naming objects. Variables appear as capital letters or strings starting with a capital letter, and constants, as strings of lower case letters. The predicates are connected together to compose sentences by means of logical connectives. In this chapter we find only two of them: the conjunction connective, or *and*, denoted '$\wedge$', and the implication connective denoted '$\rightarrow$'. Sentence 4.1, for instance, means that for whatever values of *A, M, E* and *U*, if *amount_of_material(A, M, E, U)* is true then *material(M)* and *entity(E)* and *compatible(M,E)* and *amount_of_material_unit(U)* are also true. If instead of the conjunction connective we had the disjunction connective, or *or*, denoted '$\vee$', it would mean that if *amount_of_material(A, M, E, U)* is true then at least one of *material(M), entity(E), compatible(M, E)* or *amount_of_material_unit(U)* is also true. When variables appear in logical sentences they may refer to all objects in the world or only to some object in the world. In the former case they are said to be *universally quantified*, and in the latter, *existentially quantified*. The convention in this chapter is that, unless specified otherwise, variables are universally quantified, like in sentence 4.1 above (note the expression 'for whatever values of *A, M, E* and *U*' in the explanation of its logical meaning). Existentially quantified variables are explicitly indicated like the variables *Vi, Unit, R* and *S* in sentence 4.15 (section 4.3).

Given descriptions in logic, consequences of these descriptions can be inferred according to the meaning of the logical connectives. In our case, we use inference, applying different methods as appropriate along the way, to connect metadata logical descriptions to models. Simply put, the models are consequences of the metadata descriptions and modelling heuristics rules.

4.1.1. On the Making of *Ecolingua*

To start developing *Ecolingua* we followed an informal methodology found in [Uschold and Gruninger, 1996]. A diverse data set generated by a tropical forest logging experiment in the Amazon, Brazil [Biot, 1995], was the initial case study, from which meta-level data properties were abstracted. For construction of the conceptual ontology we used the Ontolingua Server, a tool for ontology construction and sharing, based on Ontolingua, a translational approach for ontologies portability [Gruber, 1993].

The server provides automatic translation of ontologies into several target languages, including a KIF-like Prolog-readable language. Given that Prolog is our choice of language for *Ecolingua* implementation, we needed to extract a manageable and correctly encoded knowledge base from the output generated by the server's translator - a large file (5.3Mb) containing an ill-structured knowledge base composed by *Ecolingua* plus the complete ontologies from which we reused definitions. For re-engineering the server's output, we have built tools for syntactic correction, consistency checking, pruning and mapping of logical sentences into more elegantly constructed Horn clauses.

4.2. Model Structuring

Model structuring, or conceptual model formulation, is the most creative step in model development. Here the modeller maps his or her understanding of the ecological system into model elements of different types (namely, state variables, driving variables, intermediate variables, parameters, sources and sinks) and relationships among them (namely, material transfers and information transfers). Forrester diagrams are a commonly used graphic language through which this kind of system abstraction is represented ([Haefner, 1996], [Grant et al., 1997], [Ford, 1999]).

Automated model structuring is our current research focus. In this section we will present some of the formal definitions we use, finishing up with a working example.

4.2.1. Inferring Model Structure

Through the ontological information in *Ecolingua* we can describe the quantities and relations observed in data sets. The next step is to use these descriptions to guide the design of ecological models. The purpose of this section is to show the logical mechanisms we are applying to try and accomplish that. To make the discussion concrete we shall be using formal notation but, as far as possible, we shall avoid giving the details of the inference mechanisms because this would distract from the flow of the example elaborated. The formal representations themselves have also been simplified to their bare essentials, so for example we have greatly simplified the way in which heuristic rules are described and used in abductive inference in section 4.2.1.2. This keeps the example short.

We shall first give a simple example problem (section 4.2.1.1.) for which a formal representation consistent with our ontology can be given. We then set up a formal representation of the forms of modelling expertise needed to solve this problem (section 4.2.1.2.) and show how these can be applied to produce the solution (section 4.2.1.3.).

4.2.1.1. Example Problem

Modelling Scenario

Suppose a system composed of grass and deer that eat the grass (this scenario is based on the grass-deer ecosystem example found in [Haefner, 1996], chapter 3). A system dynamics model is to be constructed whose **purpose** is to simulate carbon flow between the population of grass and the population of deer. One model **assumption** is that *the per capita rate of growth of grass is constant, therefore the total growth will be the per capita rate times the total amount of carbon present in the population*. Suppose also that the modelling **database** contains measurements of the *rate of deer consumption of grass carbon* given in *g/year*.

Initial descriptions: quantity types and quantity relations

The metadata sources are the ecosystem's database, naturally, and also the model assumptions, since they may refer to quantities that are relevant to the model design. The description of quantity types and quantity relations is a manual task that consists of choosing the axioms that correctly define them. Quantity types axioms are part of *Ecolingua* - in section 4.1 we exemplified them with quantity types *amount* and *material rate*. The quantity relations axioms, in turn, are placed in the **modelling knowledge** module, since modelling is ultimately establishing relations between quantities.

In our example the model assumption is a relation between three quantities, namely, *per capita* (or *specific*) *rate of growth of grass*, *total* (or *absolute*) *growth of grass* and *amount of carbon in grass*. The axiom in the modelling knowledge module that defines the kind of quantity relation in the model assumption is the following:

$$
\begin{aligned}
inc\_total\_rate([Rspf, Amt], Rabs) \rightarrow\ & spf\_mat\_rate(Rspf, M, Eo, Ed, Uspf) \wedge \\
& constant(Rspf) \wedge \\
& amount\_of\_material(Amt, M, Ed, Uamt) \wedge \\
& abs\_mat\_rate(Rabs, M, Eo, Ed, Uabs) \wedge \\
& Rabs = Rspf * Amt \wedge \\
& Uabs = Uspf * Uamt
\end{aligned} \qquad (4.5)
$$

The logical meaning of the axiom is that if a relation of type *incremental total rate* holds between *Rspf* and *Amt* giving *Rabs* then it must be true that *Rspf* is a constant quantity of type *specific material rate* of *M* from *Eo* to *Ed* given in *Uspf,* and *Amt* is a quantity of type *amount of material* of *M* in *Ed* given in *Uamt,* and *Rabs* is a quantity of type *absolute material rate* of *M* from *Eo* to *Ed* given in *Uabs*, where *Rabs = Rspf * Amt* and *Uabs = Uspf * Uamt.* Note that what makes the total rate *incremental* is the *specific material rate* and *amount of material* quantities sharing the argument *Ed. Ecolingua* axioms 4.1 to 4.4 complete the definitions of quantity types *amount of material* and *specific* and *absolute material rates*.

According to 4.5 we can describe the quantity relation in the model assumption by stating:

qtty_desc(inc_total_rate([pc_grass_growth,grass_C],total_grass_growth)) (4.6)

And the quantity in our database, *rate of deer consumption of grass carbon*, is stated as a quantity of type *absolute material rate* according to axiom 4.2:

qtty_desc(abs_mat_rate(consumption, 'C', grass, deer, g/year)) (4.7)

Inferring Inflows

We will restrict the example to inferring *inflow* model elements - *flows* that enter a state variable. Suppose we have already inferred two state variables, *grass_C* and *deer_C*, as elements of our System Dynamics model. Formally, the state variables are asserted as:

sv_elem(grass_C) (4.8)
sv_elem(deer_C) (4.9)

Our task is to hypothesize what might be the flows into the *grass_C* and *deer_C* state variables. We formulate this question as an attempt to find a solution to the goals:

inflow_elem(F, grass_C)
inflow_elem(F, deer_C)

where *inflow_elem(F, S)* is satisfied if *F* is the name of a flow which could legitimately enter state variable *S*. To solve this goal formally we need to define the modelling principles and heuristic knowledge connecting the solutions to the problem knowledge expressed in clauses 4.6 and 4.7 above. This is done in the next section.

4.2.1.2. Representation of Modelling Expertise

One way of hypothesizing the flows in a System Dynamics model is by relating them to high-level properties of the data and modelling assumptions. This is a form of heuristic knowledge so we can never be sure that we have a complete set of definitions covering all circumstances but it appears that common situations can be covered using comparatively simple and general rules. Two examples are given below.

The first rule below states that we can infer an inflow element, *F*, into a state variable, *S*, if a *fixed proportion incremental process* relation (with rate *R*) holds between that flow and the state variable. This kind of relation also allows us to infer that *R* would be the parameter regulating the inflow *F* (for simplicity we are restricting ourselves to inferences of *inflows* only). The second allows an inflow element, *F*, into a state variable, *S*, to be inferred if an *inflow* relation, *F*, holds between an entity *E* and a material *M*, and composing *E* and *M* form the name of the state variable *S*.

$$\begin{aligned} inflow\_elem(F, S) \leftarrow\ & model\_rel(fixed\_prop\_inc\_proc(F, R, S)) \wedge \\ & sv\_elem(S) \end{aligned} \tag{4.10}$$

$$\begin{aligned} inflow\_elem(F, S) \leftarrow\ & model\_rel(inflow(F, M, E)) \wedge \\ & composition(E, M, S) \wedge \\ & sv\_elem(S) \end{aligned} \tag{4.11}$$

In material flow models, state variables represent storage of some material by some entity. The predicate *composition,* that always succeeds, simply merges the names of entities and materials, producing names of state variable candidates, e.g. *composition(deer, 'C', deer_C).*

In 4.10 *composition* is not necessary because from 4.5 *S* is of type *amount_of_material,* which is already a quantity type that suits a state variable representation (this bears on the rationale for inference of state variables, which is not included in this discussion).

We are left with *model_rel* to define. This succeeds when an appropriate high-level property (model relation) can be inferred from our problem description. Once again, this will involve heuristics and our general definition of *model_rel* sets up the conditions under which these heuristics may be introduced. The meaning of 4.12 is that we know (and describe) a quantity or relation between quantities, *Q*, whose existence would be implied by a certain model relation *Mrel*. Because *Mrel* would imply *Q* and *Q* exists we then take *Mrel* as true. This kind of inference is called *abduction*.

$$model\_rel(Mrel) \leftarrow qtty\_desc(Q) \wedge \\ model\_rel\_opt(Q \leftarrow Mrel) \tag{4.12}$$

All that is needed to know about abduction here is that it is used to characterize the hypotheses-oriented inference style of model relations. In abduction, we assume a hypothesis to be true, as opposed to proving that the hypothesis is true (deductive method). Within our metadata-supported automated modelling perspective, abduction is used to hypothesize alternative model relations given that certain quantities and relations between them hold, reflecting the fact that different interpretations of an ecological system's data may yield different models.

Finally, we need to supply the heuristic rules which allow us to infer model relations from specific quantity descriptions. The two below are sufficient for our example. The first says that an *absolute material rate*, *R*, that measures the transfer of material, *M,* originating somewhere into entity, *E*, suggests as possible model relation an *inflow*, *R*, of *M* into *E*. The second says that if we have a *total rate of increase*, *Rabs*, of amount *Amt*, with specific rate *Rspf*, this may indicate a *fixed proportional increase* model relation. Recall that we know that *M* is a material, *E* is an entity, *Amt* is an amount, etc., from *Ecolingua* definitions 4.1 to 4.4.

$$model\_rel\_opt((abs\_mat\_rate(R, M, \_, E, \_) \leftarrow inflow(R, M, E))) \tag{4.13}$$

$$model\_rel\_opt((inc\_total\_rate([Rspf, Amt], Rabs) \leftarrow fixed\_prop\_inc\_proc(Rabs, Rspf, Amt))) \tag{4.14}$$

Definitions 4.10 to 4.14 above can now be used with the problem description of 4.6 and 4.7 to solve the problem described in section 4.2.1.1. We explain in the next section how this is done.

4.2.1.3. Solving the Example Automatically

The application of our modelling expertise rules to solve the example can be visualized through the proof trees in figures 5 and 6. Figure 5 shows how the solution *inflow_elem(consumption, deer_C)* is reached satisfying the goal *inflow_elem(F, deer_C)* and figure 6, the solution *inflow_elem(total_grass_growth, grass_C)* satisfying the goal *inflow_elem(F, grass_C)*. The boxes in the trees contain expressions that have been proved. Expressions in bold boxes are given truths, either because they describe the problem (4.6, 4.7), or because we are assuming they have already been proved (4.8, 4.9), or because they are a ground clause of a predicate that is always proven true (*composition* predicate) . The expressions labelled **H** are the heuristic rules (4.13 and 4.14), with the literals that are inferred abductively appearing underlined.

4.3. Parameter Estimation - Deriving Ecological Functions' Specifications

Values for model parameters that calibrate equations are often estimated as some mathematical function applied over groups of quantities values in the database. If the model is to be backed up by a database, these groups should reflect ''natural'' data clusters, dictated by structural, spatial and/or temporal properties of the data, each kind of property at a time or in a combined manner.

We have built a mechanism that takes descriptions of data properties in *Ecolingua* and picks all the quantity values groups that are consistent with the data properties. Correspondent specifications of simple functions, namely *mean*, *minimum*, *maximum* and *summation*, can then be derived, one for each of the groups.

As an example, suppose *diameter at breast height* (DBH) is a parameter in a simulation model of timber production based on a logging experiment. The experiment database contains DBH measurements taken from different places at different times. According to the experiment's design, a sample's location is given as treatment and replicate, while sampling time is given as sampling campaign and season. Let us also suppose that for the model's purposes, a DBH mean is adequate to calibrate the DBH parameter. The question is, which DBH values in the database should be selected for the *mean* calculation? Which DBH values would produce a *mean* that best suits the predictions that the model will be asked to perform? There are many possibilities: a blind overall *mean* across all the values; temporally constrained *means* for each sampling campaign; spatially constrained *means* to each treatment; more finely grained, spatially constrained *means* to each treatment and replicate; temporally and spatially jointly constrained *means* to each treatment and sampling campaign, and so on.

The clause 4.15 below, for instance, synthesizes *mean* models constrained by a treatment, sampling campaign and threshold value condition, if any. The conceptual graph in figure 4 includes the *Ecolingua* relations used here.

$$
\begin{array}{l}
\mathit{synthesise(Qtty, Ent,} \\
\qquad \mathit{(mean\_model(Mean, St\_unit)} \leftarrow \\
\qquad\quad \mathit{standard\_unit(Qtty, St\_unit)} \wedge \\
\qquad\quad G_v = \{\mathit{Vi\_conv} \mid \text{exists } \mathit{Vi, Unit, R, S}\,. \\
\qquad\qquad\qquad \{\mathit{data\_composition(Ent, Qtty)} \wedge \\
\qquad\qquad\qquad\; \mathit{magnitude(Qtty, Vi, Unit,} \\
\qquad\qquad\qquad\qquad \mathit{location(treatment(T, replicate(R))),} \\
\qquad\qquad\qquad\qquad \mathit{time(campaign(C), season(S)))} \wedge \\
\qquad\qquad\qquad\; \mathit{number}(V_i) \wedge \\
\qquad\qquad\qquad\; \mathit{convert}(V_i, \mathit{Unit, Vi\_conv, St\_unit}) \wedge \\
\qquad\qquad\qquad\; \mathit{threshold\_bound(magnitude(Qtty, Vi\_conv, Unit,} \\
\qquad\qquad\qquad\qquad\qquad \mathit{location(treatment(T, replicate(R))),} \\
\qquad\qquad\qquad\qquad\qquad \mathit{time(campaign(C), season(S))),} \\
\qquad\qquad\qquad\qquad \mathit{Th\_cond)} \wedge \\
\qquad\qquad\qquad \mathit{Th\_cond}\}\})) \leftarrow \mathit{Mean} = \sum \mathit{Gv/\ cardinality(Gv)}
\end{array}
\tag{4.15}
$$

Suppose that:

- the standard unit for DBH is *cm*;
- *logging8* is one of the logging treatments, where eight trees with DBH > 50 cm are logged per hectare;
- *post-logging10* is one of the sampling campaigns carried out 10 years after *logging8* was applied; and
- there is a threshold value for measurements taken during the *rainy* season - only the measurements greater than 25 cm are relevant.

Using 4.15 to prove *synthesise(dbh, tree, Model)* derives as one of the *mean* models:

$$
\begin{array}{l}
\mathit{Model} = \mathit{(mean\_model(}\sum \mathit{Gv/\ cardinality(Gv), cm)} \leftarrow \\
\qquad\quad \mathit{standard\_unit(dbh, cm)} \wedge \\
\qquad\quad G_v = \{\mathit{Vi\_conv} \mid \text{exists } \mathit{Vi, Unit, R}\,. \\
\qquad\qquad\qquad \{\mathit{data\_composition(tree, dbh)} \wedge \\
\qquad\qquad\qquad\; \mathit{magnitude(dbh, Vi, Unit,} \\
\qquad\qquad\qquad\qquad \mathit{location(treatment(logging8, replicate(R))),} \\
\qquad\qquad\qquad\qquad \mathit{time(campaign(post\_logging10), season(rainy)))} \wedge \\
\qquad\qquad\qquad\; \mathit{number}(V_i) \wedge \\
\qquad\qquad\qquad\; \mathit{convert}(V_i, \mathit{Unit, Vi\_conv, cm}) \wedge \\
\qquad\qquad\qquad\; \mathit{threshold\_bound(magnitude(dbh, Vi\_conv, cm,} \\
\qquad\qquad\qquad\qquad\qquad \mathit{location(treatment(logging8, replicate(R))),} \\
\qquad\qquad\qquad\qquad\qquad \mathit{time(campaign(post\_logging10), season(rainy))),} \\
\qquad\qquad\qquad\qquad \mathit{Vi\_conv} > 25) \wedge \\
\qquad\qquad\qquad\; \mathit{Vi\_conv} > 25\}\})
\end{array}
$$

The *mean* model has the form of a first order logic sentence encompassing all the relevant constraints for grouping and operation of the tree DBH values, namely, *mean* calculation formula, conversion of all values into the standard unit of measure *cm*, treatment, campaign an season concerned and threshold value condition. Another version of the mechanism is able to calculate the function value if measurements are available.

As it is, the mechanism synthesizes exhaustive function models - the choice of which to pick is left to the modeller. An improvement to be made is to use model requirements and assumptions to restrict the synthesis. Another improvement is the handling of more complex ecological functions.

4.4. Parameter Estimation - Complex Parameter Estimation Breakdown

In section 4.3. we addressed the synthesis of models of simple ecological functions for estimation of model parameters. In this section we address a somewhat converse problem. Given that a value of a certain parameter in the model is to be estimated via a complex composite function, can data properties determine how the process of calculating this complex parameter should be broken down into simpler steps? Here we try to learn from human problem solving. Given this task, a modeller will not break down the complex parameter estimation process arbitrarily, but rather will do so with respect to the properties of the data that support the model.

Consider again the modelling scenario in section 4.3, with the database at hand containing measurements of the DBH of 360 trees taken once in 1994 and once in 1999. Again, refer to figure 4 for the *Ecolingua* classes and relations used here.

data_composition(tree, dbh) (4.16)

sampling_points_number(dbh, 360) (4.17)
sampling_year(dbh, [year(1994), year(1999)]) (4.18)

sampling_frequency(dbh, year(1994), 1)
sampling_frequency(dbh, year(1999), 1) (4.19)

Now suppose that a DBH parameter is required for the model equations and that it is to be estimated as the *mean annual increment of DBH*.

A possible process for estimation of the parameter is:

1. Calculate the total increment of DBH of each individual tree during the 5 year period between the two measurements taken:

   total(increment(dbh, Indiv)) = last_value(dbh, Indiv) - first_value(dbh, Indiv) (4.20)

2. Then, approximate the annual increment of DBH of each tree by dividing the total increment calculated above by the number of years in the period between the two measurements:

   annual(increment(dbh, Indiv)) = total(increment(dbh, Indiv))/5 (4.21)

3. And finally, have the *mean annual increment of DBH of trees* as the sum of the annual increment of all the 360 trees divided by the number of trees:

   mean(annual(increment(dbh, tree))) = sum(Indiv, 1, 360, annual(increment(dbh, Indiv)))/360 (4.22)

We have implemented a Definite Clause Grammar (DCG) that given the sentence *mean(annual(increment(dbh,tree)))* generates the sequence of equations above (in reverse order).

A DCG is a kind of formal grammar based in first order logic ([Pereira and Warren, 1980], [Abramson and Dahl, 1989]), used to generate and recognize valid sentences in a language. In our application, the grammar rules specify a successive rewriting of sentences, starting with the parameter sentence *mean(annual(increment(dbh, tree)))*. This rewriting process represents the breakdown of a complex parameter estimation into gradual simpler calculations.

Conditions in the bodies of the DCG rules tune up the sub-calculations according to the properties of the data. For instance, through the instantiation of the relation *sampling_year* to our data, the grammar can find out the time span (5 years) to approximate DBH annual increment (equation 4.21). A detailed account of the DCG and its functioning briefly discussed here can be found in [Brilhante, 1999].

In the context of an automated modelling system, a feature of this kind can assist modellers in devising estimation processes for complex parameters that are endorsed by the properties of the database behind the model.

5. A Look Ahead and Conclusion

Computer technologies have enabled the production, storage and potential accessibility of data in all domains to grow exponentially. The highly decentralized nature of these data repositories foster diverse forms of data organization, structuring and representation. Paradoxically, such diversity hinders distributed accessibility, sharing and reuse of information.

This is the motivation for the trend today of deployment of ontologies as domain models: a concise, unifying, agreed layer of relevant concepts that can comprehend data set specific semantics and, at the same time, preserve data autonomy. Over the past ten years ontologies have been abundantly produced. Libraries of ontologies are now available on the Internet. Nonetheless, widespread reuse of ontologies is a reality to be seen. Reuse is not yet considered cost-effective, demanding lots of time and effort. The number of published systematic studies on reuse of ontologies is thus low. To our knowledge [Cohen et al., 1999] is the only publication along these lines. Similarly, we hear a lot about otologies design, but we do not know much detail about what people do with ontologies, in a realistic scale, once they are designed. Three aspects, in particular, make the construction of *Ecolingua* a realistic experiment on ontologies reuse. Firstly, ecological data may involve concepts from very diverse domains. This multidisciplinary nature of the target ontology demanded reuse of terms from several ontologies. Secondly, *Ecolingua* has a well defined purpose: to provide a vocabulary and axioms for expression of ecological data properties. The ultimate goal does not lie on the design, but on the use of the designed ontology. And thirdly, we used state-of-the-art machinery available: the web-based Ontolingua server.

Ecolingua can be beneficial beyond the scope of this research project. It can be used as an inter-lingua for knowledge sharing, not restricted to the purpose of connecting metadata to model design. People involved with collection, organization and analysis of ecological data, designers of field sampling strategies and modellers are all potential users.

A great deal of modelling expertise relies on intuition and it is unlikely that this will ever be fully understood and mapped. However, there are standard routine, yet effort demanding, forms of connections between data and models that can be formalized and automated, as the results presented here start to demonstrate. One of the positive effects of using higher-level descriptions of data to guide the automation is a narrower space of model hypotheses as opposed to the large ones generated by the less informed model induction methods.

As far as ecological modelling practices and tools are concerned, we see the use of metadata-supported modelling features as a complimentary resource to be added to conventional practices, bringing about the benefit of facilitating the construction of models that are substantiated by data properties.

Acknowledgments

This research is supported by the Brazilian Government - CAPES, grant BEX1498/96-7. Thanks to the referees for their valuable suggestions.

References

[Abramson and Dahl, 1989] Abramson, H. and Dahl, V. (1989). Logic Grammars. Springer-Verlag, New York.

[Benz and Knorrenschild, 1997] Benz, J. and Knorrenschild, M. (1997). Call for a common model documentation etiquette. Ecological Modelling, 97(1,2): 141-143.

[Benz and Voigt, 1996] Benz, J. and Voigt, K. (1996). Aufbau eines Systems zur strukturierten Suche von Informatiosquellen für den Umweltschutz im Internet. In Lessing, H. and Lipeck, U. W. (ed.). *Proceedings of the Tenth Informatik für den Umweltschutz Symposium*, Hannover 1996, pages 232-241. Metropolis Verlag, Marburg.

[Biot, 1995] Y. Biot (1995). Data Survey Report - BIONTE Project, INPA - Brazil and ODA - UK. Technical Report. Manaus, Brazil.

[Bratko, 1990] Bratko, I. (1990). Prolog Programming for Artificial Intelligence. Addison-Wesley.

[Brilhante, 1999] Brilhante, V. (1999). Using Formal Metadata Descriptions for Automated Ecological Modeling. In *EDSSAI-99, Papers from the AAAI-99 Workshop on Environmental Decision Support Systems and Artificial Intelligence*, 1999, Orlando, Florida, USA, pages 90-95. Technical Report WS-99-07. AAAI Press, Menlo Park, California.

[Cohen et al., 1999] Cohen, P., Chaudhri, V., Pease, A. and Schrag, R. (1999). Does Prior Knowledge Facilitate the Development of Knowledge-based Systems? In *AAAI-99, Proceedings of the Sixteenth International Conference on Artificial Intelligence*, 1999, Orlando, Florida, USA, pages 221-226. AAAI Press, Menlo Park, California.

[Dzeroski and Todorovski, 1993] Dzeroski. S. and Todorovski, L. (1993). Discovering Dynamics. In Utgoff, P.E. (ed.). *Proceedings of the Tenth International Conference in Machine Learning*, pages 97-103. Morgan Kaufmann, San Mateo, CA.

[Dzeroski and Todorovski, 1995] Dzeroski. S. and Todorovski, L. (1995). Discovering Dynamics: from Inductive Logic Programming to Machine Discovery. Journal of Intelligent Information Systems 4:89-108.

[Dzeroski et al., 1997] Dzeroski, S., Grbovic, J., Walley, W. and Kompari, B. (1997). Using machine learning techniques in the construction of models. II. Data analysis with rule induction. Ecological Modelling 95:95-111.

[Dzeroski et al., 1999] Dzeroski, S., Todorovski, L., Bratko, I., Kompare, B. and Krizman, V. (1999). Equation Discovery with Ecological Applications. In Fielding, A. (ed.). Machine Learning Methods for Ecological Applications, pages 185-207. Klwer Academic Publishers, Boston.

[Farquhar et al., 1996] Farquhar, A., Fikes, R. and Rice, J. (1996). The Ontolingua Server: a Tool for Collaborative Ontology Construction. Technical Report KSL-96-26, Computer Science Department, Stanford University.

[Ford, 1999] Ford, A. (1999). Modeling the Environment: an Introduction to Systems Dynamics Modeling of Environmental Systems. Island Press.

[Grant et al., 1997] Grant, W., Pedersen, E. and Marín, S. (1997). Ecology and Natural Resource Management: Systems Analysis and Simulation, John Wiley & Sons.

[Gruber, 1993] Gruber, T. (1993). A Translation Approach to Portable Ontology Specifications. Knowledge Acquisition 5(2):199-220.

[Guttenplan, 1986] Guttenplan, S. (1986). The Languages of Logic. Blackwell Publishers.

[Haefner, 1996] Haefner, J. (1996). Modeling Biological Systems: Principles and Applications. Chapman & Hall.

[Hand, 1997] Hand, D. (1997). Intelligent Data Analysis: Issues and Opportunities. In Liu, X., Cohen. P. and Berthold., M. (ed.) Advances in Intelligent Data Analysis. Lecture Notes in Computer Science 1280. *IDA-97, Proceedings of the First International Symposium on Intelligent Data Analysis*, pages 1-14. Springer-Verlag, Berlin, Heidelberg.

[Hilborn and Mangel, 1997] Hilborn, R. and Mangel, M. (1997). The Ecological Detective: Confronting Models with Data. Princenton University Press.

[Hoch et al., 1998] Hoch, R., Gabele, T. and Benz, J. (1998). Towards a standard for documentation of mathematical models in ecology, Ecological Modelling 113: 3-12.

[Karalic, 1992] Karalic, A. (1992). RETIS: A Knowledge Acquisition System. User's manual. Software version: 2.07/nf. Ljubljana.

[Kompare et al., 1994] Kompare, B., Bratko, I., Steinman, F. and Dzeroski, S. (1994). Using machine learning techniques in the construction of models. I. Introduction. Ecological Modelling 75/76:617-628.

[Krizman et al., 1995] Krizman, V., Dzeroski, S. and Kompare, B. (1995). Discovering Dynamics from Measured Data. Electrothecnical Review 62:191-198.

[Kubat et al., 1997] Kubat, M., Bratko, I. and Michalski, R. (1997). A Review of Machine Learning Methods. In Michalski, R., Bratko, I. and Kubat, M. (ed.). Machine Learning and Data Mining: Methods and Applications, pages 3-69. John Wiley & Sons, New York.

[Lemmon, 1987] Lemmon, E. (1987). Beginning Logic (2nd edition). Chapman and Hall.

[Lorenz et al., 1989] Lorenz, G., Uhrmacher, A., Simon, K. and Bossel, H. (1989). Application of Artificial Intelligence Methods to the Representation and Modelling of Tree Growth. In *Proceedings of the IUFRO Conference, Artificial Intelligence and Growth Models for Forest Management*, 1989,Vienna, Austria.

[Muetzelfeldt et al., 1989] Muetzelfeldt, R., Robertson, D., Bundy, A. and Uschold, M. (1989). The Use of Prolog for Improving the Rigour and Accessibility of Ecological Modelling. Ecological Modelling 46:9-34.

[Pereira and Warren, 1980] Pereira, F. and Warren, D. (1980). Definite Clause Grammars for Language Analysis. Artificial Intelligence 13:231-278.

[Rickel and Porter, 1997] Rickel, J. and Porter, B. (1997). Automated modeling of complex systems to answer prediction questions. Artificial Intelligence Journal 93(1-2):201-260.

[Robertson et al., 1991] Robertson, D., Bundy, A., Muetzelfeldt, R., Haggith, M. and Uschold, M. (1991). Eco-Logic: logic-based approaches to ecological modelling. The MIT Press.

[Sterling and Shapiro, 1994] Sterling, L. and Shapiro, E. (1994). The Art of Prolog: Advanced Programming Techniques (2nd edition). The MIT Press.

[Todorovski and Dzeroski, 1997] Todorovski, L. and Dzeroski, S. (1997). Declarative Bias in Equation Discovery. In *Proceedings of the Fourteenth International Conference on Machine Learning*, pages 376-384, Morgan Kaufmann, San Mateo, CA.

[Uhrmacher, 1995] Uhrmacher, A. (1995). Reasoning about changing structure: a modeling concept for ecological systems. Applied Artificial Intelligence 9:157-180.

[Uschold, 1991] Uschold, M. (1991). The use of domain information for comprehension and construction of simulation models. DAI Research Paper 534, Department of Artificial Intelligence, University of Edinburgh.

[Uschold and Gruninger, 1996] Uschold, M. and Gruninger, M. (1996). Ontologies: principles, methods and applications. The Knowledge Engineering Review 11(2):93-136.

[Uschold et al., 1984] Uschold, M., Harding, N., Muetzelfeldt, R. and Bundy, A. (1984). An intelligent front end for ecological modeling. DAI Research Paper 223, Department of Artificial Intelligence, University of Edinburgh.

[Xia and Smith, 1996] Xia, S. and Smith, N. (1996). Automated Modelling: a Discussion and Review. The Knowledge Engineering Review 11(2):137-160.