Towards Compositional Modelling of Ecological Systems via Dynamic Flexible Constraint Satisfaction

Jeroen Keppens and Qiang Shen
School of Artificial Intelligence, University of Edinburgh
80 South Bridge, Edinburgh EH1 1HN, UK
E-mail: {jeroen,qiangs}@dai.ed.ac.uk

Abstract

This paper presents an initial work towards the development of a technique for compositional modelling of ecological systems. A knowledge representation framework is devised to suit system dynamics - the underlying modelling paradigm adopted. Model fragment selection and composition is achieved by means of dynamic flexible constraint satisfaction problem (DFCSP) solution techniques. A method for automatically translating a scenario and a model fragment library into a dynamic constraint satisfaction problem (DCSP) is proposed. From this DCSP all consistent scenario models can be derived through a hierarchy of subproblems and different model evaluation criteria can be imposed through specific sets of constraints. A technique is then suggested to order the modelling choices within each of the DCSP subproblems, in terms of the degree of model detail, and hence the preferences associated with these choices. The result is a compositional modeller in which the tasks of maintaining model consistency, selecting model boundary, evaluating model suitability are accomplished by resolving a single DFCSP.

Introduction

The present work aims at a compositional modelling (Falkenhainer, B. & Forbus, K.D. 1991) approach for ecological systems. This application domain poses several new challenges to compositional modelling. Ecological systems consist of an overwhelming amount of components and processes. Yet, ecologists are mostly interested in aggregate concepts and aggregate interactions within an eco-system. Therefore, ecological models consist of abstract processes representing aggregate phenomena of interest. In certain problem domains, such as hydro-ecology (Heller, U. & Struss, P. 1996), these aggregate processes are equivalent to the combined effect of the relevant first principles and in other domains, such as population dynamics, the aggregate processes merely attempt to approximate behaviour associated with certain phenomena.

Conventional compositional modellers derive the requirements for an adequate model from an initial state specification or some task specification (Keppens, J. & Shen, Q. 2000). Approaches such as (Farquhar 1993), aim at extrapolating all possible states of a model from the initial state and do not take alternative modelling assumptions into account. A task specification or query determines which variables must be related to one another and some compositional modellers, such as (Levy, A.Y., Iwasaki, Y., & Fikes, R. 1997; Nayak, P.P. & Joskowicz, L. 1996), search for the simplest model that establishes this relation. However, these approaches are geared towards engineering applications and use knowledge that is specific to that domain, e.g. topological device structures.

Alternatively, the scope of the model and the required level of detail may be computed in terms of the underlying factors, such as the granularity of the time scale at which significant changes occur (Rickel, J. & Porter, B. 1997). However, in many domains of ecology, such as population dynamics, concise first principles and the assumptions underlying any approximations are not readily available due to the intrinsic complexity of the phenomena. For example, there are no known mathematical laws underlying the phenomenon of predation between two populations. Instead, an empirical or artificial relation that is assumed to approximate the unknown underlying first principles is used. Unfortunately, the assumptions on which such approximations are based, e.g. adequate time scale, are not completely understood.

This paper proposes an alternative approach to compositional modelling that addresses these issues. A knowledge representation framework is introduced that enables the representation of a space of ecological models for a given scenario and that allows phenomena and modelling alternatives to be distinguished and composed. The problem of model composition is described as a dynamic flexible constraint satisfaction problem (DFCSP). A set of requirements of models to be constructed are translated into a combination of hard and preference constraints imposed over emerging partial models. The modified local repair techniques (Miguel, I. & Shen, Q. 2000) are adopted to guide the search for finding a solution to such DFCSPs, i.e. an adequate composed ecological model.

Background

Ecological modelling with system dynamics

Many different kinds of modelling approach are used in ecology. A most common paradigm is *system dynamics* (Forrester 1961). In system dynamics, the phenomena of interest are represented as levels and flows between them. The change per time unit in a level equals the total of inflows minus the total of outflows. Additional variables and influences describe the relations

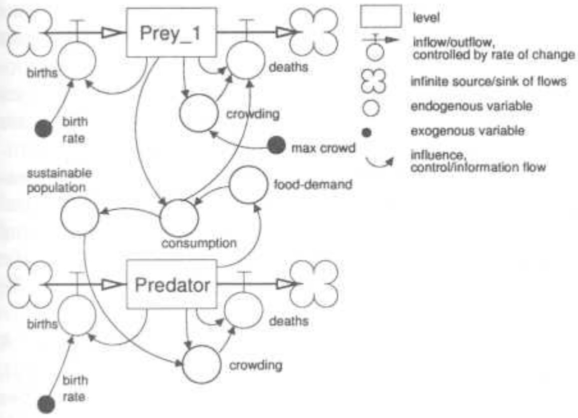

Figure 1: A system dynamics model of predation

between the levels and flows. As such, system dynamics provides an interface to modelling with differential equations and allows features of other paradigms to be integrated (Robertson, D. *et al.* 1991).

To illustrate the system dynamics approach to ecological modelling in the present work, the following scenario is used:

population(`predator`) $\wedge$ population(`prey_1`) $\wedge$
population(`prey_2`) $\wedge$ feeds-on(`predator`, `prey_1`) $\wedge$
feeds-on(`predator`, `prey_2`) $\wedge$ competition(`prey_1`, `prey_2`)

It describes a world consisting of three populations - a `predator` population that feeds on the other populations `prey_1` and `prey_2`. The latter two populations compete with one another over scarce resources.

The concepts representing ecological systems do not normally play one of a few well-defined roles in a model. Instead, depending on the scenario and the problem at hand, various phenomena are considered with respect to the concepts. In the present scenario, reproduction within both populations, the predation behaviours of `predator` with respect to the prey population and competition between `prey_1` and `prey_2` may be of relevance. In other scenarios, different phenomena such as infection of diseases may also have to be considered.

Many models of these different phenomena exist. Consider the predation phenomena involving `prey_1`, of which figure 1 shows one possible model. A kind of growth phenomenon conceptualising changes in population size is necessary for both populations. This is because predation affects the change in the level of the prey population since predation kills prey, as well as the change in the level of the predator population since the total amount of available prey affects the sustainable population.

In figure 1, the growth phenomenon is represented by a level, an inflow and an outflow for both populations. If more information were available about the population (e.g. a more specific type of population) a more precise growth phenomenon would be modelled. Based on these concepts of population growth, the specific models of growth can be added. In this case a simple linear reproduction model is used that is limited by a maximal sustainable population level. The model of the predation behaviour, limits the capacity of the predators based on the available prey. The total consumed prey is added to the total outflow of prey. This description is represented in figure 1 by variables and influences between them. If necessary, new exogenous variables could be added to explain the current exogenous variables.

Dynamic and flexible constraint satisfaction

The simplest of constraint satisfaction problems (CSP) can generally be specified as a triplet $\langle X, D, C \rangle$ where X is a set of attributes $\{x_1, \ldots, x_n\}$, D is a set of domains $\{D_1, \ldots, D_n\}$ describing the potential values of the attributes and C is a set of constraints relating some of the attributes. Each attribute $x_i \in X$ must be assigned a single value $d \in D_i$. Such attribute assignments will be denoted as $x_i : d$. Each $c(x_i, \ldots, x_j) \in C$ specifies a subset D^c of $D_i \times \ldots \times D_j$ such that $\forall (d_i, \ldots, d_j) \in D^c, c(d_i, \ldots, d_j)$ is consistent with c. The purpose of solving a CSP is to find a tuple $(d_1, \ldots, d_n)$ such that the attribute assignments $x_1 : d_1, \ldots, x_n : d_n$ cause all constraints in C to hold.

Although CSPs have been studied in great detail, they are not sufficiently equipped to deal with two features that are frequently present in real-world problems. On the one hand, the problem specification can be dynamic in nature. This implies that the CSP specification changes over time (Miguel, I. & Shen, Q. 1999) or with respect to other attribute assignments (Mittal, S. & Falkenhainer, B. 1990). On the other hand, constraints do not necessarily impose hard requirements. For example, certain attribute assignments may be preferred over others. The former type of CSP is referred to as dynamic CSP (DCSP) and the latter to flexible CSP (FCSP).

Similar to the work presented in (Mittal, S. & Falkenhainer, B. 1990), this work requires the handling of DCSPs in which the set of relevant attributes is defined by other attributes. In order to solve such CSPs, activity predicates are introduced such that:

$$\forall x_i \in X : \mathrm{active}(x_i) \leftrightarrow \vee_{d \in D_i} x_i : d$$

¿From this it follows that $\neg\mathrm{active}(x_i)$ implies that no attribute assignment is considered for x_i. The traditional CSP constraints, termed compatibility constraints, are still applicable in these DCSPs. A compatibility constraint $c(x_i, \ldots, x_j)$ is translated as $c(x_i, \ldots, x_j) \vee \neg\mathrm{active}(x_i) \vee \ldots \vee \neg\mathrm{active}(x_j)$. As a result, the determination of the truth of an activity predicate implicitly results in a set of constraints as well. In addition to compatibility constraints, the dynamic CSP also contains so-called activity constraints. These come in the form of implications where the consequent consists of a literal containing the activity predicate of one of the attributes.

When flexible constraints are allowed, the satisfaction of a constraint becomes a matter of degree. The degree of satisfaction (of attribute assignments) with respect to a constraint, say, $c(x_i, \ldots, x_j)$ may be defined by an elastic relation $R : D_i \times \ldots \times D_j \mapsto L : (d_i, \ldots, d_j) \rightarrow s_R(d_i, \ldots, d_j)$, where L is a satisfaction scale (e.g. $[0, 1]$) (Miguel, I. & Shen, Q. 2000). In this way, the original notions of constraint satisfaction and violation is extended such that the degree to which different assignments satisfy a constraint can be ordered. In the present work, a method is proposed to define such an ordering of preferences for alternative attribute assignments depending on their underlying role in model composition.

Knowledge representation

The representation of model fragments in this work mainly follows the general framework of the Compositional Modelling Language (CML) (Bobrow, D. *et al.* 1996), apart from certain syntactical differences. The system being modelled is described by a scenario. A scenario is denoted by a pair $\langle O, \Phi \rangle$ where $O = \{o_1, \dots, o_n\}$ is a set of object constants, called the scenario objects, and Φ is a set of relations $\phi(o_i, \dots, o_j)$ over the object constants. An example of an ecological scenario is provided above. The required compositional modeller should translate a scenario into a system dynamics model, given a knowledge base and a task specification.

The knowledge base

The knowledge base used by a compositional modeller largely consists of a model fragment library which is a collection of predefined model fragments. Each model fragment represents a way of modelling a particular feature of some system or subsystem under certain conditions. By selecting and instantiating a set of model fragments, a compositional modeller constructs models of a system satisfying the prescribed task specification.

A model fragment μ is a tuple $\langle P^s, P^t, \Phi^s, \Phi^t, A \rangle$ where $P^s(\mu) = \{p_1^s, \dots p_m^s\}$ is a set of variables, called *source-participants*, $P^t(\mu) = \{p_1^t, \dots, p_n^t\}$ is a set of variables, called *target-participants*, $\Phi^s(\mu) = \{\phi_1^s, \dots, \phi_v^s\}$ is a set of relations, called *structural conditions*, whose free variables are elements of P^s, $\Phi^t(\mu) = \{\phi_1^t, \dots, \phi_s^t\}$ is a set of relations, called *postconditions*, whose free variables are elements of $P^s \cup P^t$, $A(\mu) = \{\alpha_1, \dots, \alpha_t\}$ is a set of relations, called *assumptions*, such that for $i = 1, \dots, s$:

$$\forall p_1^s, \dots, \forall p_m^s, \exists p_1^t, \dots, \exists p_n^t \; \phi_1^s \wedge \dots \wedge \phi_v^s \rightarrow (\alpha_1 \wedge \dots \wedge \alpha_t \rightarrow \phi_i^t)$$

The source and target participants in a model fragment are variables representing domain objects. These objects may be entities or subsystems of the real-world system of interest. Alternatively, they may be conceptual entities that, when instantiated, assume the role of variables within a scenario model, thus representing significant properties of the system. Instantiated participants of the latter type are called *model variables*. The relations existing between the objects represented by participants are defined by the conditions in the model fragments. For each participant, at least one unary relation is defined that specifies the type of the participant.

The structural conditions of a model fragment describe the subsystem setting to which that model fragment is applicable. A model fragment μ with $P^s(\mu) = \{p_1^s, \dots p_m^s\}$ is said to be *applicable* with respect to the set of object constants $o_1, \dots, o_m$ of a database Δ if for each $\phi \in \Phi^s(\mu)$, $\phi_{o_1^s/p_1, \dots, o_m^s/p_m}$ holds in Δ. Here, a database stands for a collection of domain object constants and known relations between them.

The assumptions in the above definition are relations on source-participants and object constants and they are used to represent specific features of the models that include the associated model fragments. For example, assumptions may indicate the inclusion of certain phenomena or distinguish between alternative ways of modelling these phenomena. Their truth depends on the specific requirements of the model and hence, they can not be deduced merely from the scenario.

In this work, the assumptions come in two types. A *relevance assumption*, denoted as $\text{relevant}(h, p_1, \dots, p_q)$, states that the associated model fragment describes a phenomenon h, which applies to the participants $p_i, \dots, p_j$. They are considered with respect to certain object constants $o_1, \dots, o_q$ that instantiate $p_1, \dots, p_q$. The tuple $(h, o_1, \dots, o_q)$ is called the *subject* of a relevance assumption. If the entity type of a participant p is denoted by $T(p)$, then the *specificity* of a relevance assumption can be defined as $(T(p_1), \dots, T(p_q))$. Relevance assumptions with the same subject, but different specificity are inconsistent with one another.

Consider, for example, the model fragment: $\text{population}(p) \wedge \text{relevant}(\text{growth}(p)) \rightarrow \text{level}(l) \wedge \text{unit-of}(l, \text{population}) \wedge \text{rate}(r) \wedge \text{size-of}(p, l) \wedge \text{flow}(r, \text{source-sink}, l)$. This fragment introduces all objects that are required to represent the phenomenon $\text{growth}(p)$: the growth rate r and the population level l, which represents the size of p, and a relation between them $\text{flow}(r, \text{source-sink}, l)$, which can be translated as $\frac{d}{dt}\mathbf{l} = \mathbf{r}$. An alternative model fragment could be defined for a specific type of population for which a model of population growth exists that is more suitable than the default. For example, the growth of a population of holometabolous insects should be modelled by four levels instead of one, representing the appropriate stages of growth: egg, larva, pupa and adult.

A *model assumption* $\text{model}(\phi, t)$ states that the associated model fragment represents the source-participant or structural condition ϕ in a specific way described by t. For each database that contains $\text{model}(\phi, t)$ it is said that the *model type* of ϕ is t. Such assumptions are used to distinguish between different ways of describing (or explaining) object constants or relations between object constants. The object constant or relation ϕ between object constants to which the model assumption is instantiated is called the *subject* of the model assumption. Model assumptions with different model types and the same subject are inconsistent with one another.

Consider, for example, the model fragment: $\text{rate}(r) \wedge \text{level}(l) \wedge \text{unit-of}(l, \text{population}) \wedge \text{flow}(r, \text{source-sink}, l) \text{model}(\textit{exponential}, r) \rightarrow \text{birth-rate}(r_0) \wedge (r = r_0 \times l)$. This fragment contains the "exponential" model type for the number of births per time unit r of a population (being $r = r_0 \times l$). Another model fragment could contain an alternative model type, say "logistic"(Roughgarden 1996), model type for r.

The target-participants represent new objects that should be added to the scenario model when the model fragment is applicable and the assumptions are deemed true. The postconditions define new relations between source-participants, target-participants or a combination of both. They are either an assignment or a predicate.

Assignments are only defined for model variables. An assignment $p_i = I(p_k, \ldots, p_l)$ defines participant p_i as a function of participants $p_k, \ldots, p_l$, $i \notin \{k, \ldots, l\}$. This function may be composed of standard arithmetic operations or may be the compositional operators (see (Bobrow, D. *et al.* 1996)). In the latter case, the postconditions are partial definitions that can be combined with other postconditions that describe other assignments to the same model variable for p_i.

A predicate $R(p_k, \ldots, p_l)$ either adds new information about the scenario model or represents a predefined set of assignments. The latter case is called an assignment predicate. An example of this is $\text{flow}(r, l_1, l_2)$, where r is a participant of the rate type and l_1 and l_2 are source-participants of the level type, and which represents that $\frac{d}{dt}l_1 = -r$ and $\frac{d}{dt}l_2 = r$.

Additional representation formalisms

Most other supportive representation formalism of CML exist in the proposed compositional modeller as well. Some participants belong to predefined types. Entity type definitions are used to specify these. Each entity has a number of entity attributes representing features of particular importance (e.g. population size).

The entity types are organised in a type hierarchy. A subclass inherits all entity attributes and other features of its superclasses and may add new ones. For instance, the holometabolous-insect-population type is a subclass of the population type. In applications of ecological modelling, the entity attributes specific to the more specialised classes may be related to entity attributes introduced in its parent classes. In particular, certain aggregation relations are currently defined in this way.

In order to compare between different emerging models, alternative model assumptions may be assigned preferences. In the absence of any preference order, the model types are considered to be equally suitable (assuming that all other features are equal). Further preference orders may be assigned for certain subtypes of the participants in the subject of the model assumptions. For example, in addition to a preference order between model types of the reproduction of a population, an alternative order can be defined for a subpopulation, of which the behaviour should differ from that normally seen by default. Relevance assumptions with the same subject are not explicitly ordered in the knowledge base as the specificities of the individual assumptions already form a preference ordering.

Task specification

A range of ecological models can be specified using the knowledge representation formalisms discussed above. Which of the possible models is most appropriate depends on the requirements of the task at hand. These are prescribed in the task specification. Because, in the general case, ecological models do not entirely follow from first principles, little knowledge on the adequacy of partial models can be derived from a knowledge base. Therefore, the task specification must impose additional criteria on the model. In the present version of this work, the following criteria are contained in the task specification:

- The *objectives* that establish the features that must be included in the model. In system dynamics, certain domain variables are required to be simulated to show their potential behaviour. Hence, these variables must be included as entity attributes of entity type definitions. In what follows such variables, explicitly required as the modelling objectives, are called *objective variables*.
- The maximal level of *model complexity*, which is herein expressed in terms of the total number of variables because this is one of the simplest ways of comparing model complexity between models covering different sets of phenomena (Rickel, J. & Porter, B. 1997).
- The *specific requests* which dictate certain model variables to be exogenous or to be simply excluded from any resulting model. Such model variables are named *requested variables*.

The compositional modelling approach presented herein aims at constructing a model that represents the phenomena that affect the objective variables in as much detail as possible given the constraint on model complexity and the specific requests.

Model composition

Inference from a scenario

Through the implications represented by the model fragments in the knowledge base, new information can be derived from the given scenario description. The set of relations Φ in a database $\Delta = \langle O, \Phi \rangle$ may be pattern matched with the structural conditions of some model fragment μ by a substitution σ that maps each source-participant in $P^s(\mu)$ onto an object constant of O. Also, σ instantiates the assumptions $A(\mu) = \{\alpha_1, \ldots, \alpha_t\}$ to a set of ground relations. If $\sigma\alpha_1 \wedge \ldots \wedge \sigma\alpha_t$ is consistent with Δ, a new database $\Delta' = \langle O', \Phi' \rangle$ logically follows from Δ. O' contains every $o \in O$ and a new object constant for each $p \in P^t(\mu)$, thereby forming a new substitution σ'. Φ' is the union of Φ, the instantiated assumptions $\sigma' A(\mu)$ and the instantiated postconditions of μ, $\sigma' \Phi^t(\mu)$. The pseudo-code in figure 2 illustrates how a knowledge base of model fragments is instantiated based on a given scenario.

Such a model space is represented as a hypergraph θ. The nodes of θ are objects or relations between objects. Each hyperarc links a set of nodes $n_1, \ldots, n_p$ to another node n_q, representing $n_1 \wedge \ldots \wedge n_p \rightarrow n_q$. Initially, the nodes are those objects and object relations provided in the given scenario. For each model fragment whose source-participants and structural conditions match the objects and relations already in θ, a set of new nodes and hyperarcs are created as the results of such matches and these are added to θ. Overall, this procedure is similar to the approach presented in (Falkenhainer, B. & Forbus, K.D. 1991).

```
θ:=new hypergraph;
foreach o ∈ scenario do
  add-node(θ, o);
foreach match(μ, θ)
  justf:=empty set;
  foreach a ∈ A(μ) do
    newnode:=add-node(θ, a);
    justf:=justf∪{newnode};
  end-foreach;
  foreach p ∈ P^s(μ) do
    justf:=justf∪{find-node(θ,p)};
  foreach φ ∈ Φ^s(μ) do
    justf:=justf∪{find-node(θ,c)};
  foreach p ∈ P^t(μ) do
    o:=add-node(θ, p);
    add-justification(θ, o,justf);
  end-foreach;
  foreach φ ∈ Φ^t(μ) do
    o:=add-node(θ, c);
    add-justification(θ, o,justf);
  end-foreach;
end;
```

Figure 2: Generation of a model space

The model space in the running example will at least contain parts of models representing natural reproduction in the absence of other species for the predator, prey_1 and prey_2 species, parts of models describing the predation of prey_1 and the predation of prey_2 by predator and parts of models on the competition between prey_1 and prey_2. Figure 3 shows part of the model space that is generated from this sample scenario. From this partial model space, two types of models of natural reproduction of the predator species can be generated. Figure 3 also illustrates part of the possible predation models involving the predator and one of the prey species.

Construction of a dynamic CSP

Having derived a model space from a given scenario, individual models can be created by selecting a set of assumptions. To support this selection, information contained in this model space is translated into the description of a DCSP. This translation allows the use of the solution techniques developed for DCSPs to be employed for the selection of a consistent model. The next subsection will then show how preference orderings can be added to such DCSPs and explain how this aids in guiding the search for an adequate model.

A model is selected from the model space, presuming that a certain number of assumptions be true and that all others be false. Within the model space, each assumption represents one modelling decision from a set of mutually exclusive options. This is syntactically equivalent to an attribute and its unary domain constraint in a CSP. Hence, an attribute is created for each assumption subject in θ.

In addition to the assumptions, information on the variables that have been included in the model is also important. Any given task specification may restrict which variables are included in the model and whether they can be exogenous or endogenous, or may limit the total number of variables. Therefore an attribute is included for each instantiation of each participant that is a model variable. The domains of these attributes are {exogenous, endogenous}, representing the different roles the associated variables may play in the scenario model. In what follows, $\sigma_{\text{CSP}}(\theta)$ denotes the substitution that maps the assumption subjects and model variables to attributes and that maps the specificities, model types and variable roles to attribute assignments.

Activity constraints define the conditions under which attributes are active. The relevance of a phenomenon at a certain specificity can only be considered with respect to a set of objects to which the phenomenon is applied. Also, a model type of a certain object constant or relation can only be considered under the conditions that the object constant or the relation exists. Therefore, an attribute representing an assumption is active only if its associated subject is instantiated. The conditions under which an attribute x_i is instantiated are hereafter denoted by $\gamma(x_i)$. If x_i represents an assumption with a subject that contains the relations or participants $\phi_1, \ldots, \phi_n$ then, $\gamma(x_i) = \gamma(\phi_1) \wedge \ldots \wedge \gamma(\phi_n)$.

The $\gamma(\phi_i)$ can be computed with respect to four different cases. Firstly, if ϕ_i is an object constant or relation given in the scenario, then $\gamma(\phi_i) = \top$. Secondly, if ϕ_i is mapped onto an attribute - more specifically $\sigma_{\text{CSP}}(\theta)\phi_i = x_j$ - then $\gamma(\phi_i) = \text{active}(x_j)$. Thirdly, if ϕ_i is mapped onto an attribute assignment - that is $\sigma_{\text{CSP}}(\theta)\phi_i = x_j : d_{jk}$ - then $\gamma(\phi_i) = x_j : d_{jk}$. And finally, in all other situations, $\gamma(\phi_i)$ can be derived from the model space by tracing back each of the hyperarcs leading to ϕ_i. Each hyperarc introduces a new conjunction of assumption participants and/or relations ϕ for which $\gamma(\phi)$ can be computed in one of the other three ways just described.

The conjunctions produced for the different hyperarcs are disjoined. Suppose that the resulting expression is denoted as $\Gamma(x_i)$. The following activity constraint can then be constructed for x_i: $\text{active}(x_i) \rightarrow \Gamma(x_i)$. $\Gamma(x_i)$ is only a necessary constraint because none of associated assumptions have to be selected for the purpose of model construction. If x_i represents a model variable, then $\Gamma(x_i)$ is computed in much the same way, except that the aforementioned fourth case is checked first.

The compatibility constraints dictate what combinations of individual assumptions are consistent, and what are not. Certain combinations of assumptions may result in inconsistent relations. Inconsistent relations are caused by the instantiation of non-composable postconditions. A set of postconditions are said to be non-composable if they are assignment postconditions with the same assignee and the assigned mathematical relations cannot be combined by means of composable operators. Hence, for each pair ϕ_i, ϕ_j of non-composable postconditions instantiated in the model space, a compatibility constraint $\neg(\gamma(\sigma_{\text{CSP}}\phi_i) \wedge \gamma(\sigma_{\text{CSP}}\phi_i))$ is created.

Additional constraints govern whether a model variable is exogenous or endogenous. If $\{\phi_1, \ldots, \phi_n\}$ is the set of all assignments in the model space that have the variable v as their assignee, the following two constraints are added to the DCSP: σ_{CSP} endogenous(v)

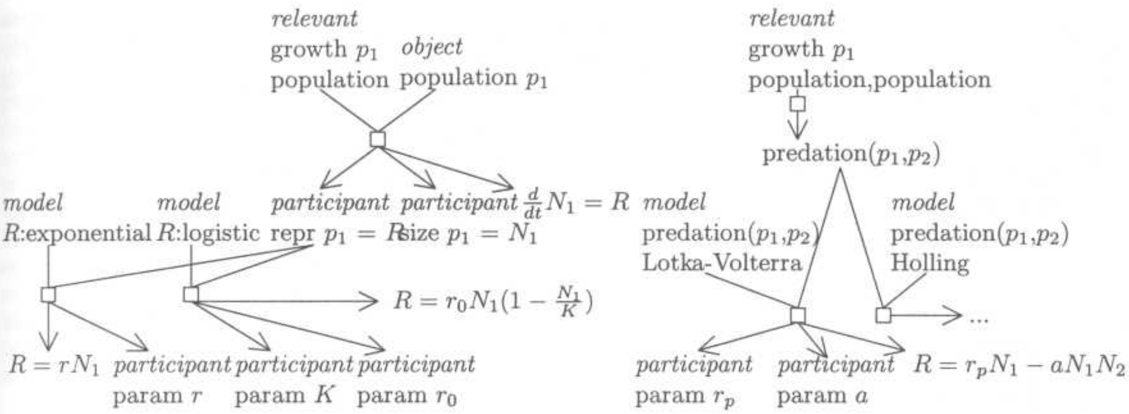

Figure 3: Partial model space for the predator prey scenario

$\leftarrow$ active($\sigma_{\mathrm{CSP}}v$) $\wedge$ ($\sigma_{\mathrm{CSP}}\phi_1 \vee \ldots \vee \sigma_{\mathrm{CSP}}\phi_n$) and σ_{CSP}exogenous(v) $\leftarrow$ active($\sigma_{\mathrm{CSP}}v$) $\wedge$ $\neg$endogenous(v). Restrictions on the activity or roles of a model variable in the task specification now map straightforwardly to the DCSP. Finally, a special purpose attribute is added that counts the total number of active model variables. This attribute is constrained by an upper limit, which is the maximal model complexity as specified in the task specification.

Using this method, the model space of figure 3 can be translated into a DCSP. A graphical representation of the constraints resulting from the partial model space given in figure 3 is shown in figure 4. Assumptions with the same subject are grouped into a single attribute. For example, the assumptions "model R:exponential" and "model R:logistic" of figure 3 are grouped into the attribute x_2 with domain {"exponential", "logistic"}. The activity constraints over the attributes representing assumptions are defined by the conditions that activate the subject. The activity constraints of attributes are defined by tracing the necessary justification in the model space. For example, x_2 represents a set of assumptions and therefore depends on its subject, which is in this case a model variable that has another CSP variable associated with it. Hence, the activity of the latter CSP variable is a necessary condition for the activity of x_2 as illustrated in figure 4.

The causal relations between the model variables are captured by the assignment postconditions. For instance, the relation $R = rN_1$ results in a compatibility constraint that states that if r is active, R must be endogenous.

Non-composable postconditions are prohibited by compatibility constraints. For example, figure 3 shows that the assignments x_4 : Lotka-Volterra and x_2 : logistic in the CSP both result in an assignment postcondition with respect to R, which has been prohibited by a compatibility constraint in figure 4.

Now, each consistent assignment in the DCSP can then be translated into a scenario model. For example, the assignment (x_1 : population, R : endogenous, N_1 : endogenous, x_2 : logistic, r_0 : exogenous, K : exogenous) is consistent with the CSP of figure 4. By considering the assumptions associated with these attribute assignments as facts, and by following the justifications defined through the hyperarcs in the model space, a model is then constructed. In particular, the following statements can be inferred: $\frac{d}{dt}N_1 = R$, $R = r_0 N_1(1 - \frac{N_1}{K})$, exogenous($r_0$), exogenous($K$).

Model selection and preference orderings

The CSP solution algorithm used is very similar to that presented in (Miguel, I. & Shen, Q. 2000). Initially, only the attributes that are not restricted by activity constraints may be active and can be assigned. Certain assignments may cause the necessary conditions for the activation of other attributes to be met, which yields a new assignment problem (essentially a new CSP) with respect to the newly added variables. The construction of such new CSPs continues until no additional necessary conditions for activation are met or until no consistent assignment is possible. In the latter case, the algorithm tries to re-assign the attributes of the previous subproblem that have led to the inconsistency. An existing local repair technique is used to perform the re-assignment, such that consistency is regained.

In this way, the DCSP is organised into a hierarchy of smaller CSPs. Each attribute assignment leads to the construction of a new CSP (if new attributes can be activated), to a solution for the DCSP or to an inconsistency. Each assignments corresponds to a set of modelling choices that affect the resulting scenario model and its level of detail. It is required that this level of detail be sufficient, yet, formal knowledge to evaluate this is not readily available in ecological modelling. As they are not related to the mechanics that caused these observations in the first place, it is, in general, difficult to deduce when adding certain features has a significant impact to the behaviour that is extrapolated from the scenario model.

The approach taken herein is that additional detail consisting of relevant model variables and relations is generally preferred as long as it does not conflict with the other requirements. Attribute assignments are therefore evaluated with respect to the amount of detail associated with the respective set of assumptions that is instantiated according to the attribute assignment. As

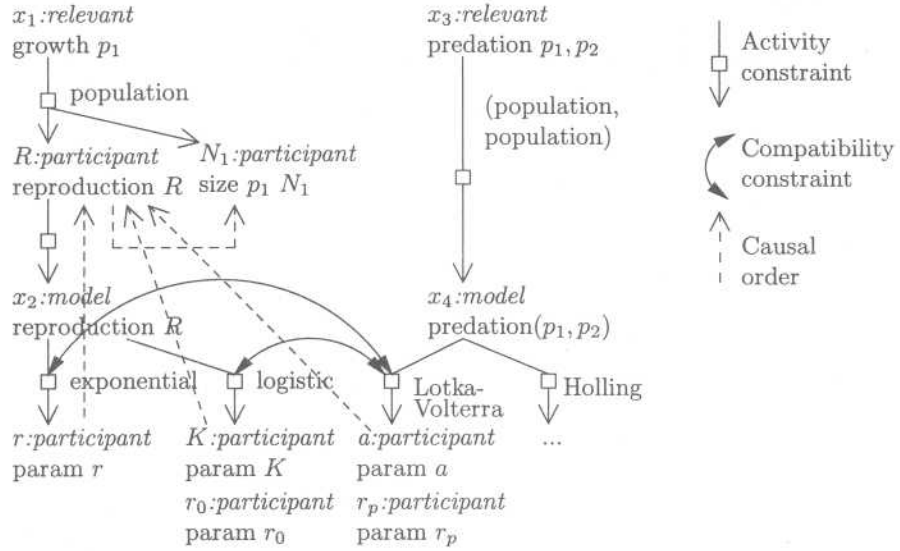

Figure 4: Graphical representation of a sample CSP for model composition

explained earlier, the knowledge base may contain orderings of assumptions with the same subject in terms of the level of detail, and hence an ordering of the preference for the associated attribute assignments.

The available preference orderings apply to the domains of individual attributes, assuming that model variables and relations whose existence depends upon the associated assumptions add relevant detail to the scenario model. These model variables and relations are regarded to adding relevant detail if they causally explain one or more of the objective variables given in the task specification.

A preference ordering is assigned to the alternative assignments within a single CSP in the DCSP hierarchy, by combining the preference orderings for individual attributes extracted from the knowledge base with the objective variables of the task specification and the causal relations between model variables. In what follows, $\Pi(\overrightarrow{x} : \overrightarrow{d_j})$ represents a preference range $[\beta_1, \beta_2]$ within which the preference of the entire assignment is contained and $\rho(x_i : d_{ij})$ denotes the maximal number of objective variables that can be causally explained by one of the attribute assignments which is enabled by the assignment $x_i : d_{ij}$.

Given a set of attributes $x_1, \ldots, x_n$, two assignments $\overrightarrow{x} : \overrightarrow{d_1} = (x_1 : d_{11}, \ldots, x_n : d_{n1})$ and $\overrightarrow{x} : \overrightarrow{d_2} = (x_1 : d_{12}, \ldots, x_n : d_{n2})$ are compared in terms of detail. Because relevance assumptions normally have a far greater impact on the scenario model than model assumptions (as model assumptions reflect modelling decisions with respect to a part of a phenomenon instead of an entire phenomenon), attribute assignments are first compared solely in terms of relevance assumptions. Suppose that the attributes $x_1, \ldots, x_j$ represent relevance assumptions and that the CSP variables $x_{j+1}, \ldots, x_n$ represent model assumptions. If $\rho(x_1 : d_{11}) + \ldots + \rho(x_j : d_{j1}) > \rho(x_1 : d_{12}) + \ldots + \rho(x_j : d_{j2})$ then $\Pi(\overrightarrow{x} : \overrightarrow{d_1}) > \Pi(\overrightarrow{x} : \overrightarrow{d_2})$. That is, the lower bound of $\Pi(\overrightarrow{x} : \overrightarrow{d_1})$ is greater than the upper bound of $\Pi(\overrightarrow{x} : \overrightarrow{d_2})$. Otherwise, if $\rho(x_1 : d_{11}) + \ldots + \rho(x_j : d_{j1}) = \rho(x_1 : d_{12}) + \ldots + \rho(x_j : d_{j2})$, the preference orderings of the individual assignments of $x_1, \ldots, x_j$ are compared. In case d_{i1} is an assignment that is preferred over d_{i2} (in other words, $x_i : d_{i1}$ represents a relevance assumption of higher specificity than $x_i : d_{i2}$) for $i = 1, \ldots, j$, then $\Pi(a\overrightarrow{x} : \overrightarrow{d_1}) > \Pi(\overrightarrow{x} : \overrightarrow{d_2})$.

If no total order can be established between two assignments based on the relevance assumptions (such that one's lower bound is greater than another's upper bound), a preference ordering is established based on the assignments of the attributes representing model assumptions. The maximal number of objective variables affected by each CSP variable x_i that represents a model assumption (i.e. $i = j + 1, \ldots, n$) is computed as $\rho(x_i) = \max_{d_{ik} \in D_i}(\rho(x_i : d_{ik}))$. This reflects the intuition that model assumptions on subjects that affect more of the objective variables should be awarded a relatively higher priority. Consider again two attribute assignments that need to be compared, over individual variables. Those individual variable assignments with the highest $\rho(x_i)$ are compared first. If no order is established on this basis, the assignments with the next highest ρ value are compared, and so on.

Using the task specification, the most appropriate model is searched. For example, suppose that the objective variables are the population sizes of `predator` and `prey_1`, that the scenario model may not contain the reproduction rate r of these two populations as an exogenous variable (both size and reproduction rate being entity attributes specified in the entity definition) and that there is a restriction on the maximum number of variables in the resulting model (although this last requirement is not actually useful for the present simple example).

This is achieved by assigning preference orderings to alternative sets of assumption assignments. Initially, 6 attributes representing relevance assumptions may be activated. These represent the following phenomena: growth of `predator`, `prey_1` and `prey_2`, predation of `prey_1` and `prey_2` and competition between the prey species. The preferred assignment is the one which activates all 6 attributes, because such an assignment will lead to the highest ρ value possible. The resulting ρ value is 6, contributed by the fact that the relevance of the growth phenomena of `predator` and `prey_1` each affect one objective variable, the predation phenomena between both affects two objective variables and the competition phenomenon and predation between `predator` and `prey_2` each affect a single objective variable.

All resulting models containing the competition phenomenon or the predation phenomenon between `predator` and `prey_2` require the growth phenomenon of `prey_2`. Therefore they involve many additional variables. If the number of additional variables is too high, the next best alternative excluding either of those phenomena is the assignment which renders the phenomena including `prey_2` inactive. This assignment still has a ρ value of 4.

Suppose that the latter (i.e. the growth phenomena of `predator` and `prey_1` and the predation phenomenon between the two are considered relevant) is chosen. Then, a new assignment problem is formulated with three attributes: two representing model assumptions for reproduction of `predator` and `prey_1` and one representing model assumptions for the predation relation. In this new CSP, not all attribute assignments can be combined. For example, none of the assignments of x_2 can be combined with assignment x_4 : `Lotka-Volterra` as demonstrated in figure 4. The preference ordering method will first evaluate the assignment of x_4 because, as opposed to the other attributes, the consequences of this assignment will result in model variables that causally explain the two objective variables. Therefore, variable assignments for the current CSP that include an assignment to x_4 (the model type for predation) are preferred over those that do not. The assignment x_2 : exponential would be prohibited anyway since it can only lead to the assignment r : exogenous which was restricted in the task specification.

Conclusion and future work

This paper has presented an initial work towards the development of a technique for compositional modelling of ecological systems. A knowledge representation framework is devised that supports system dynamics, which is commonly adopted for eco-modelling, and its typical modelling assumptions. The work allows a more elaborate task specification than conventional compositional modelling approaches, for the use of explicit criteria for model evaluation. Currently, the task specification can take a set of model objectives, restrictions on the exogenous and endogenous variables involved in the scenario model and the limit on the overall complexity of the model.

Dynamic flexible constraint satisfaction techniques are used to guide the search for a model that meets prescribed specifications. A method for automatically translating a given scenario, task specification and a model fragment library into a dynamic constraint satisfaction problem is proposed. From the resulting DCSP description all consistent scenario models that meet the task specification can be derived via a hierarchy of subproblems. Information on the preference orderings between modelling choices can be exploited to order the CSP variable assignments in these subproblems. An existing solution technique is applied to resolve such DFCSPs, in order to obtain an adequate and consistent scenario model.

A number of issues concerning ecological systems modelling remain, however. A common feature of ecological models in system dynamics is the disaggregation of a model variable, representing a population of individuals, into a set of model variables, each representing a subpopulation. Currently, each possible disaggregation has to be modelled explicitly by a model fragment and distinguished by a specific model type for the model variable in question. However, the knowledge of how to model disaggregation could be generalised, depending on the entity attributes (e.g. age, sex, social status) that define the subpopulations and on the factors that affect migration between subpopulations. It is very interesting to extend the DFCSP-based work to allow the representation of and reasoning with such knowledge.

The DFCSP solution techniques currently employed are based on the conventional local repair technique to solve DCSPs. This technique has the advantage of being able to compute a new solution based on an existing one when the constraints of the problem change. However, its use in the present work focuses only on the revision when resulting models of different complexity, or of different boundary, are required. If a problem may involve further complicated modifications to a given scenario or task specification (such as changes to model objectives), how to exploit this technique to repair the existing scenario model, without the need to resolve a new DFCSP from scratch, remains as an important future work. Last, but not least, the important issues of space and time complexity of the proposed approach to DFCSP based compositional modelling need to be studied in detail.

Acknowledgements

The first author is supported by a scholarship of the Faculty of Science and Engineering of the University of Edinburgh. Both authors are grateful to Ian Miguel and Robert Muetzelfeldt for useful discussions.

References

Bobrow, D.; Falkenhainer, B.; Farquhar, A.; Fikes, R.; Forbus, K.; Gruber, T.; Iwasaki, Y.; and Kuipers, B.J. 1996. A compositional modeling language. In *Proceedings of the 10th International Workshop on Qualitative Reasoning about Physical Systems*, 12–21.

Falkenhainer, B., and Forbus, K.D. 1991. Compositional modeling: finding the right model for the job. *Artificial Intelligence* 51:95–143.

Farquhar, A. 1993. *Automated Modeling of Physical Systems in the Presence of Incomplete Knowledge.* Ph.D. dissertation, University of Texas at Austin.

Forrester, J. 1961. *Industrial Dynamics.* MIT Press.

Heller, U., and Struss, P. 1996. Transformation of qualitative dynamic models - application in hydro-ecology. In *Proceedings of the 10th International Workshop on Qualitative Reasoning about Physical Systems*, 83–92.

Keppens, J., and Shen, Q. 2000. On compositional modelling. Revised version submitted for journal publication.

Levy, A.Y.; Iwasaki, Y.; and Fikes, R. 1997. Automated model selection for simulation based on relevance reasoning. *Artificial Intelligence* 96:351–394.

Miguel, I., and Shen, Q. 1999. Hard, flexible and dynamic constraint satisfaction. *Knowledge Engineering Review* 14(3):199–220.

Miguel, I., and Shen, Q. 2000. Dynamic flexible constraint satisfaction. *Applied Intelligence.*

Mittal, S., and Falkenhainer, B. 1990. Dynamic constraint satisfaction problems. In *Proceedings of the 8th National Conference on Artificial Intelligence*, 25–32.

Nayak, P.P., and Joskowicz, L. 1996. Efficient compositional modeling for generating causal explanations. *Artificial Intelligence* 83:193–227.

Rickel, J., and Porter, B. 1997. Automated modeling of complex systems to answer prediction questions. *Artificial Intelligence* 93:201–260.

Robertson, D.; Bundy, A.; Muetzelfeldt, R.; Haggith, M.; and Uschold, M. 1991. *Eco-Logic. Logic-Based Approaches to Ecological Modelling.* MIT Press.

Roughgarden, J. 1996. *Theory of Population Genetics and Evolutionary Ecology: An Introduction.* Prentice Hall.