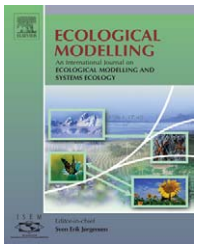
available at www.sciencedirect.com

SCIENCE DIRECT®

journal homepage: www.elsevier.com/locate/ecolmodel



Inductive revision of quantitative process models

Nima Asgharbeygi[a], *Pat Langley*[a,*], *Stephen Bay*[a], *Kevin Arrigo*[b]

[a] *Computational Learning Laboratory, CSLI, Stanford University, Stanford, CA 94305, USA*
[b] *Department of Geophysics, Mitchell Building, Stanford University, Stanford, CA 94305, USA*

ARTICLE INFO

Article history:
Available online 20 December 2005

Keywords:
Process models
Differential equations
Scientific discovery
Model revision

ABSTRACT

Most research on computational scientific discovery has focused on developing an initial model, but an equally important task involves revising a model in response to new data. In this paper, we present an approach that represents candidate models as sets of quantitative processes and that treats revision as search through a model space which is guided by time-series observations and constrained by background knowledge cast as generic processes that serve as templates for the specific processes used in models. We demonstrate our system's ability on three different scientific domains and associated data sets. We also discuss its relation to other work on model revision and consider directions for additional research.

© 2004 Published by Elsevier B.V.

1. Introduction and motivation

Most research on computational scientific discovery (e.g., Langley, 2000) has emphasized the generation of entirely new models to describe or explain data. However, scientists spend much of their time not developing new accounts of phenomena but rather revising and improving an existing model that already has credibility. If we desire computational tools that help scientists in practice, we should examine seriously issues that arise in the revision of scientific models.

There are other excellent reasons for focusing on model revision rather than generation. Most approaches to scientific discovery carry out search through the space of hypotheses or models, which can be quite large. By starting from an existing candidate, we can reduce the effective size of this space, making some tasks tractable that would otherwise be too difficult. This approach also places constraints on the search mechanism that can reduce variance and thus decrease chances of overfitting the data. Finally, model revision gives the domain user more control over the region of the model space explored, and increases the chances that he will find the new model comprehensible.

In this paper, we report one approach to the problem of scientific model revision. Unlike most earlier work on this topic, we focus on the modification of quantitative process models, a representation of knowledge that we believe offers additional advantages related to search and interpretability. We review this formalism briefly in the section that follows, along with RPM, an algorithm that alters an initial process model in response to time-series data. After this, we report experimental results on three environmental domains that involve a protist predator–prey system, water behavior in a Danish Fjord, and phytoplankton growth in an Antarctic sea. In closing, we discuss related work on model revision and outline directions for future research.

2. Process models and their revision

In this section, we review our previous work on process models and their induction from time-series data and background knowledge. After this, we present our approach to revising models within the process-modeling framework.

* *Corresponding author*. Fax: +1 650 494 1588.
E-mail address: langley@csli.stanford.edu (P. Langley).

0304-3800/$ – see front matter © 2004 Published by Elsevier B.V.
doi:10.1016/j.ecolmodel.2005.10.008

2.1. *Quantitative process models*

Scientific models are often stated formally as sets of equations, but they are also described informally in terms of the processes that determine those equations. We have developed the formalism of *quantitative process models* to encode both aspects of scientific knowledge. In this framework, a model consists of a set of processes, each of which specifies one or more equations that represent causal relations among variables. These are cast as algebraic equations for instantaneous effects or differential equations for changes over time. Processes can also include threshold conditions on variables that characterize when they are active.

A process model specifies not only a set of processes, but also the variables which they connect. A given variable may be labeled as observable, meaning it is present in the data, or it may play the role of a theoretical term that serves mainly to link processes. Each variable may also be labeled as exogenous, in that it influences other variables but is not influenced in return, or as endogenous, which means it appears in the left-hand side of one or more equations. Our framework is a quantitative variant of Forbus' (1984) qualitative process theory, from which we have borrowed many ideas.

Table 1 shows a process model for the aquatic ecosystem of the Ross Sea in Antarctica, which is based on an earlier differential equation model developed by the biolocial oceanographer member of our team (Arrigo et al., 2003). The model includes three observable terms—the exogenous variable light and the endogenous concentrations of phytoplankton (phyto) and nitrate, a nutrient. Theoretical terms that are unobservable but play key roles include detritus (dead organic matter), r_max (the maximum growth rate), n_to_c_ratio (the nitrogen to carbon ratio), and four variables (growth_rate, nitrate_rate, light_rate, remin_rate) that determine the rates of certain processes.

Table 1 – A quantitative process model for the Ross Sea ecosystem

```
model Ross_Sea_Ecosystem;
variables phyto, detritus, nitrate, light, growth_rate, nitrate_rate,
          light_rate, n_to_c_ratio, r_max, remin_rate;
observable phyto, nitrate, light;
exogenous light;
process phyto_loss;
  equations d[phyto, t, 1] = −0.1 ∗ phyto;
            d[detritus, t, 1] = 0.1 ∗ phyto;
process phyto_growth;
  equations d[phyto, t, 1] = growth_rate ∗ phyto;
process phyto_uptakes_nitrate;
  equations d[nitrate, t, 1] = −1 ∗ n_to_c_ratio ∗ growth_rate ∗ phyto;
process growth_limitation;
  equations growth_rate = r_max ∗ min(nitrate_rate, light_rate);
process nitrate_availability;
  equations nitrate_rate = nitrate/(nitrate + 5);
process light_availability;
  equations light_rate = light/(light + 50);
process global_parameters;
  equations n_to_c_ratio = 0.251;
            r_max = 0.194;
            remin_rate = 0.0676;
```

Table 2 – The process model for the Ross Sea ecosystem translated into the traditional notation of differential and algebraic equations

$$d[\text{phyto}, t, 1] = -0.1 * \text{phyto} + \text{growth\_rate} * \text{phyto}$$
$$d[\text{detritus}, t, 1] = 0.1 * \text{phyto}$$
$$d[\text{nitrate}, t, 1] = -1 * \text{n\_to\_c\_ratio} * \text{growth\_rate} * \text{phyto}$$

$$\text{growth\_rate} = \text{r\_max} * \min(\text{nitrate\_rate}, \text{light\_rate})$$
$$\text{nitrate\_rate} = \text{nitrate}/(\text{nitrate} + 5)$$
$$\text{light\_rate} = \text{light}/(\text{light} + 50)$$

$$\text{n\_to\_c\_ratio} = 0.251$$
$$\text{r\_max} = 0.194$$
$$\text{remin\_rate} = 0.0676$$

The first process in the model characterizes loss of phyto due to miscellaneous sources (e.g., grazing and sinking), along with an increase in detritus, whereas the second specifies the rate of change for phyto as a function of its current concentration and its growth_rate.[1] The third process concerns the decrease in nitrate due to uptake by phytoplankton. Next, growth_limitation indicates that growth_rate is the maximum rate r_max times the minimum of two theoretical terms for growth limitation, nitrate_rate and light_rate, which are functions of nitrate and light availability.

A final "process" specifies the values of three parameters that occur across other processes. This has no causal interpretation, but it does clarify that quantities like the nitrogen to carbon ratio and maximum growth rate must be the same throughout the model. This will prove important later, when we consider techniques for revising models in response to ecological and environmental data.

We can utilize a process model of this sort, together with initial values, to simulate the system's behavior over time and thus predict values for each endogenous variable. We have implemented a module that transforms a given model into a set of algebraic and ordinary differential equations, as shown in Table 2, after which it uses standard computational techniques to solve these equations and generate predicted trajectories for the variables. The only nonstandard issues that arise involve checking processes to determine whether their conditions are satisfied and, if not, excluding their influence on those time steps. Later, we present trajectories that a revised version of this model predicts for phyto and nitrate, along with observations for the same variables.

Process models provide an explanation of observations in that they offer a causal account in terms of processes and equations that are familiar to domain specialists. For example, Table 3 presents some generic processes relevant to aquatic ecosystems that serve as background knowledge. These differ from *specific* processes in that they do not commit to particular variables or parameter values. However, they can indicate constraints, such as stating that the variable *P* in the generic process grazing must have type p_species and that its coefficient gamma must fall between 0 and 1. The table also states that the same parameter must appear in multiple equations within some processes.

[1] The notation d[*X*, *t*, 1] here refers to the first derivative of *X* with respect to time.

Table 3 – Some generic processes for aquatic ecosystems with type constraints on their variables and range constraints on their parameters

```
generic process grazing;
  variables P{prey_species}, Z{pred_species},
            R{detritus}, G{graze_rate};
  parameters gamma [0, 1];
  equations d[P, t, 1] = −1.0 ∗ G ∗ Z;
            d[R, t, 1] = gamma ∗ G ∗ Z;
            d[Z, t, 1] = (1 − gamma) ∗ G ∗ Z;

generic process Ivlev_rate;
  variables G{graze_rate}, P{prey_species};
  parameters delta[0, 10], rho[0, 10];
  equations G = rho ∗ (1 − exp(−1 ∗ delta ∗ P));

generic process nutrient_remineralization;
  variables N{nutrient}, M{remin_rate},
            R{detritus}, NtoC{n_const};
  equations d[N, t, 1] = M ∗ NtoC ∗ R;

generic process exponential_growth;
  variables P{prey_species}, G{grow_rate};
  equations d[P, t, 1] = G ∗ P;
generic process uptakes_nutrient;
  variables N{nutrient}, G{grow_rate},
            P{prey_species}, NtoC{n_const};
  equations d[N, t, 1] = −1 ∗ NtoC ∗ G ∗ P;

generic process growth_limitation;
  variables G{grow_rate}, MAX{r_const},
            NR{n_rate}, LR{l_rate};
  equations G = MAX ∗ min(NR, LR);

generic process detritus_loss_to_remin;
  variables R{detritus}, M{remin_rate};
  equations d[R, t, 1] = −1 ∗ M ∗ R;
```

In a previous paper (Langley et al., 2003), we proposed the task of inducing process models like the one in Table 1 from time-series data and from background knowledge like the generic processes in Table 3. We noted that this task differs from those typically studied in machine learning, in that process models characterize the behavior of dynamical systems with continuous variables that change over time, and thus are not independently and identically distributed. Moreover, such models are explanatory in nature, in that processes themselves are not observable, processes can interact to produce complex behavior, and process models can include theoretical variables that are also unobservable. However, these complicating factors are offset by the assumption that the dynamical systems are deterministic (although observations may contain noise), since scientists often make this assumption.

We have also made arguments, which we will not repeat here, that existing methods for machine learning and knowledge discovery do not solve the task of inductive process modeling. In order to address this task we developed an initial algorithm, called IPM, that carries out exhaustive search through the entire space of process models. The system then selects the parameterized model which produces the best score on an evaluation criterion that incorporates both error and model complexity.

Experiments with IPM produced encouraging results on real-world data collected from batteries on the Space Station, but our studies with environmental models, which had motivated our research on process model induction, dealt only with synthetic data and involved a target model with only five processes. The availability of both new environmental data sets and codification of additional ecosystem processes has encouraged us to extend the IPM framework and evaluate its behavior in this new context.

2.2. *The RPM revision algorithm*

Although IPM produced promising results, it had drawbacks that limited its applicability. The system constrained its search space by utilizing background knowledge about generic processes, but the space of models could still be large. Also, IPM provided no way to guide the search toward models a scientist might find more plausible. As argued earlier, model revision seems an appropriate response to both issues, so we developed an extended system, RPM, that adopts this approach to process model induction. This revision module is a key component in an integrated environment that we are developing to aid scientists in developing and improving their models. This environment assumes that models are cast as sets of quantitative processes and that generic versions of these processes are available as background knowledge.

The RPM algorithm requires the user to specify four inputs. These include: an initial model that encodes beliefs about the processes that are most likely involved; a set of constraints representing acceptable changes to the initial model that specify which initial processes should be fixed, can be removed, or have their parameters changed; a set of generic processes that may be added to the initial model; and observations to which the revised model should be fit. The initial model constitutes the user's best guess about the processes that are present in the system, whereas the allowed changes indicate his areas of uncertainty. Combined with the candidate additions, these provide RPM with a heuristic that guides search toward parts of the space that are consistent with domain knowledge.

As output, the algorithm generates a set of revised models that are sorted by their distance from the initial model and presented with their mean squared error on the training data. The distance between a revised model and the initial model is defined as the number of processes that are present in one but not in the other. This output format lets one observe the trade-off between performance of revised models and their similarity to the initial model, leaving the user to determine the best compromise between the factors and to select an appropriate model from those in the suggested set.

The RPM system operates in two main stages. The first involves searching through the model space and finding all model structures that are consistent with the specified constraints, including user-approved changes. The system first generates all instantiations of the user-recommended generic processes that satisfy constraints on variable types; these become candidates for addition to the model. Next, RPM carries

out search through the space of model structures, using the initial model as the start state. The search method utilizes two operators: adding a process from the set of instantiated generic processes and removing a process from the initial model. The current implementation uses breadth-first search so that models closer to the initial model are considered first. The algorithm also performs sanity checks on each candidate that ensure it forms a single connected graph and includes all observable variables. The result is a set of revised model structures that attempt to explain relations among the variables.

The second stage determines, for each model structure, the parameter values for new processes and ones allowed to change. To this end, RPM utilizes a combination of the Levenberg–Marquardt method (Levenberg, 1944; Marquardt, 1963) interleaved with randomized jumps. The search algorithm starts by selecting a random initial point that falls within the parameter ranges specified in the generic processes. The algorithm then attempts to optimize the parameters with the Levenberg–Marquardt routine until it converges to a local optimum. RPM then generates several new candidates by positing random jumps along the dimensions of the parameter vector. If a jump leads to lower error, it moves to that point and returns to the Levenberg–Marquadt method; otherwise, the system repeatedly generates new candidates and gradually increases the jump size. However, if RPM observes no improvement after 20 iterations, it restarts the entire process from a new random initial point. We have found that this parameter-fitting method gives enough flexibility to produce reasonable matches to time series from a variety of domains.

3. Experimental evaluation of RPM

Naturally, we were interested in how RPM behaves in practice on actual modeling problems. In this section, we report our experience with three distinct domains with different characteristics, ranging from purely physical to ecological processes and taken from both experimental and observational settings. We also compare RPM's results to those produced by the earlier IPM system, which constructs models from generic processes rather than revising an initial candidate.

3.1. *Predator–prey interactions in protists*

Within ecology, models of predator–prey systems are among the simplest in terms of the number of variables and parameters involved, making them good starting points for our evaluation. In particular, the protist system composed of the predator *Didinium nasutum* and the prey *Paramecium aurelia* is well known in population ecology, and Jost and Adiriti (2000) report time-series data for this system, recovered from an earlier report by Veilleux (1976), that are now available on the World Wide Web. These include measurements for the two species' populations at 12-hour intervals, as shown later. The data are fairly smooth, with observations at regular intervals and several clear cycles.

Table 4 presents an initial model for this two-species system that includes three processes. One such process, nasutum_decay, states that the population of the predator nasutum decreases as a direct function of the current population size. An analogous relationship, aurelia_decay, posits that a similar relation produces decreases in the prey population but involves a different parameter. The third process, aurelia_exponential_growth, claims that the prey also grows over time at a rate that more than offsets decay. Of course, this model lacks a crucial feature, in that it includes no process for predation, which we have omitted for the purposes of demonstration.

Table 4 – A simple process model for a predator–prey ecosytem

```
model Predator_Prey;
variables nasutum, aurelia;
observable nasutum, aurelia;
process nasutum_decay;
  equations d[nasutum, t, 1] = −1 ∗ 1.2 ∗ nasutum;
process aurelia_decay;
  equations d[aurelia, t, 1] = −1 ∗ 0.5 ∗ aurelia;
process aurelia_exponential_growth;
  equations d[aurelia, t, 1] = 2.5 ∗ aurelia;
```

As noted earlier, before RPM can improve an incomplete model of this sort, the user must provide a set of generic processes it can use to this end. Table 5 shows some processes that we extracted from our reading of the Jost and Adiriti article. Again, each generic process specifies one or more generic variables with type constraints (in braces), a set of parameters with ranges for their values (in brackets), and a set of algebraic or differential equations that encode causal relations among the variables. Each process can also include one or more conditions, although none appear in this example.

The table shows four such generic processes. The first-two—logistic_growth and exponential_growth—characterize the increase in a species' population in an environment that has unlimited resources, but they differ in their precise functional forms. The other two processes—predation_holling and

Table 5 – A set of generic processes for predator–prey models

```
generic process logistic_growth;
  variables S{prey};
  parameters psi [0, 3], kappa [0, 1];
  equations d[S, t, 1] = psi ∗ S ∗ (1 − kappa ∗ S);

generic process exponential_growth;
  variables S{prey};
  parameters beta [0, 2];
  equations d[S, t, 1] = beta ∗ S;

generic process predation_volterra;
  variables S1{prey}, S2{predator};
  parameters pi [0, 1], nu [0, 1];
  equations d[S1, t, 1] = −1 ∗ pi ∗ S1 ∗ S2;
            d[S2, t, 1] = nu ∗ pi ∗ S1 ∗ S2;

generic process predation_holling;
  variables S1{prey}, S2{predator};
  parameters rho [0, 1], gamma [0, 1], eta [0, 1];
  equations d[S1, t, 1]= − 1 ∗ gamma ∗ S1 ∗ S2/(1+rho ∗ gamma ∗ S1);
            d[S2, t, 1]= eta ∗ gamma ∗ S1 ∗ S2/(1 + rho ∗ gamma ∗ S1);
```

Table 6 – Best revised process model for the predator–prey ecosytem

```
model Predator_Prey;
variables nasutum, aurelia;
observable nasutum, aurelia;
process nasutum_decay;
  equations d[nasutum, t, 1] = −1 ∗ 1.057 ∗ nasutum;
process aurelia_logistic_growth;
  equations d[aurelia, t, 1] =
                1.943 ∗ aurelia ∗ (1 − 0.000579 ∗ aurelia);
process nasutum_aurelia_holling;
  equations d[aurelia, t, 1] = −1 ∗ 0.0329 ∗ aurelia ∗
                nasutum/(1 + 0.0126 ∗ 0.0329 ∗ aurelia);
            d[nasutum, t, 1] = 0.294 ∗ 0.0329 ∗ aurelia ∗
                nasutum/(1 + 0.0126 ∗ 0.0329 ∗ aurelia);
```

predation_volterra—describe alternative forms of feeding that produce an increase in the number of predators and a decrease in the prey, again differing only in the forms of their equations. All four processes are generic in the sense that they do not commit to specific variables.

We ran RPM on the Veilleux time series, telling it to retain the nasutum_decay process from Table 4 but to consider removing the other two processes and to consider adding processes from Table 5. We also told it to improve the parameters in any processes that were retained. The resulting search space contained 26 model structures, which RPM took about 3 hours to examine on a Linux PC with a 2.8 GHz Pentium 4 processor. Table 6 presents the revised model with the best score, which lacks the original process aurelia_decay, replaces the exponential growth process for this organism with one for logistic growth, and adds a new predation process.

Fig. 1 shows the trajectories observed for both species, along with those generated by the best revised model. The mean squared error is 340.584 for *D. nasutum* and 2390.537 for *P. aurelia*, which is substantially better than the errors produced by the initial model structure with improved parameters. More important, the theoretical curves track the heights and timing of the observed trajectories quite well. RPM appears to account for the major behavioral features of this ecosystem in ecologically plausible terms.

However, we should note that these results are based on only a portion of Veilleux's data. The first ten days of measurements have considerably lower peaks, suggesting that a different regime is operating. RPM was unable to find a model that could reproduce the entire time series accurately, which led us to provide it with the reduced set of observations. Thus, we do not yet have a complete explanation of these data, which indicates either that that the data were influenced by unknown factors, that our model space omitted some important processes, or that our revision system can still be improved.

3.2. *Water dynamics in Ringkøbing Fjord*

The Ringkøbing Fjord, on the Danish west coast, is a shallow body of water that is fed by tributaries from the land and that exchanges water with the North Sea through a narrow channel on its western edge. A barrier with 14 gates has been constructed across this channel in order to regulate water flow between the Fjord and the sea. Officials would like to predict in advance the water level at the gates, so they can be ready to open or close them as needed.

Our treatment of this domain borrows from work by Todorovski (2003), who reports that domain experts specified a partial dynamic model. They hypothesized that, when the gates are closed, the dominant influences on the water level *H* inside the gates to the Fjord are the wind direction Wdir and wind speed Wvel, as well as the inflow Q of fresh water per second. When the gates between the Fjord and the sea are open, the difference between *H* and sea level (hsea) is also an important factor. Following Todorovski, we have incorporated all of these influences into the initial model presented in Table 7.

The domain experts maintained that wind affects height, but they did not provide a specific function for this relationship. As shown in the table, we included the process windForcingSine to serve this role in our initial model, but we also provided RPM with the four other generic processes shown in Table 8. The first incorporates the cosine of wind direction in an effort to capture influences of the wind's other components. Three other processes reflect the idea that, as wind pushes water to one side of the Fjord, gravitational potential energy builds up in the water and causes it to resist. Our treatment is similar to that reported by Todorovski, although we believe our processes provide clearer physical accounts than the polynomial functions he utilized.

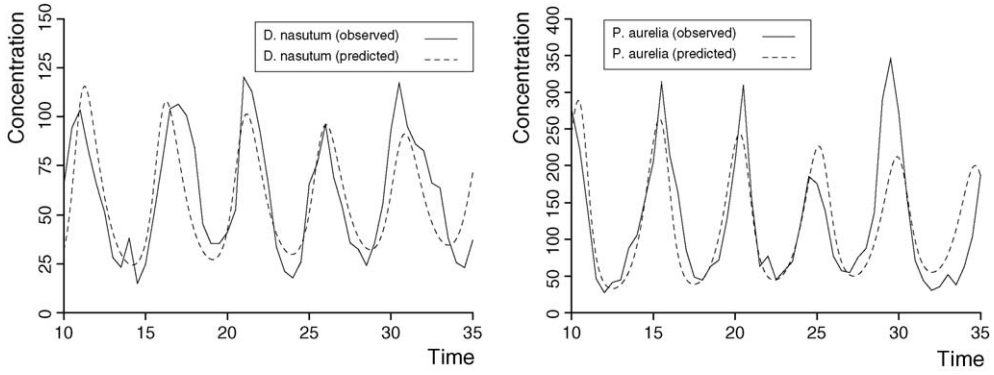


Fig. 1 – Concentrations of the predator *D. nasutum* (left) and the prey *P. aurelia* (right) as measured by Veilleux and predicted by the revised model in Table 6.

Table 7 – Initial process model for Ringkøbing Fjord

```
model Fjord_Height_Dynamics;
variables H, hsea, n, Q, Wdir, Wvel, WaterFlow, hf, hw;
observable H, hsea, n, Q, Wdir, Wvel;
exogenous hsea, n, Q, Wdir, Wvel;
process waterFlowThroughGates;
  equations WaterFlow = −1000 ∗ 0.5 ∗ n ∗ (H − hsea);
process freshWaterInput;
  equations WaterFlow = Q;
process flowHeightRelation;
  equations d[hf, t, 1] = 86400 ∗ WaterFlow/A(H);
process windForcingSine;
  equations d[hw, t, 1] = 0.05 ∗ Wvel ∗ sin(Wdir);
process totalHeight;
  equations H = hf + hw;
```

We had access to almost one year's observations of the Fjord, sampled every 5 hours, for all of the variables in the initial process model. To evaluate RPM in this domain, we treated the first 1100 observations as a training set and used the remaining 551 data points as a test set to measure generalization error. We provided the initial model, the generic processes, and the training data to RPM, which searched the revision space and returned the best model at each distance from the initial one. The system considered 32 distinct model structures, which took about 12 CPU hours.

The best-scoring model on the training set included two processes not in the initial version. One of these, wind_forcing_cosine, posits that the east-west component of the wind, measured in polar notation, has an effect on the water height. The other, wind_forcing_simple_damping, incorporates resistance to the wind force due to the gravitational force. The resulting model had a mean squared error of 0.0056725 on the training data and 0.0099752 on the test data. Equally important, it appears to be reasonable physically, which of course was encouraged by the generic processes we provided.

This model's predictions for water height are shown in Fig. 2, with points to the left of the vertical line denoting the training set and those to its right the test set. As the figure shows, RPM predicts the qualitative behavior of the Fjord dynamics in a reasonable way. In particular, the trajectory tracks the high peaks of the water height very well. These results offer evidence that the system can revise a model in ways that explain the training set and also extrapolate effectively to new observations. Todorovski also reports encouraging results with his LaGramge discovery system, although our results are not directly comparable because he used ten-fold cross validation, rather than testing the induced model's extrapolative ability.

Table 8 – Additional generic processes for Ringkøbing Fjord

```
generic process wind_forcing_cosine;
  variables Wvel{speed}, Wdir{direction}, h{sublevel};
  parameters b[−0.1, 0.1];
  equations d[h, t, 1] = b ∗ Wvel ∗ cos(Wdir);

generic process wind_forcing_cosine_damped;
  variables Wvel{speed}, Wdir{direction}, h{sublevel};
  parameters b[−0.1, 0.1], c[−0.1, 0.1];
  equations d[h, t, 1] = b ∗ (c ∗ Wvel ∗ cos(Wdir) − h);

generic process wind_forcing_sine_damped;
  variables Wvel{speed}, Wdir{direction}, h{sublevel};
  parameters b[−0.1, 0.1], c[−0.1, 0.1];
  equations d[h, t, 1] = b ∗ (c ∗ Wvel ∗ sin(Wdir) − h);

generic process wind_forcing_simple_damping;
  variables h{sublevel};
  parameters c[0, 1], k[0, 1];
  equations d[h, t, 1] = −k ∗ (h − c);
```

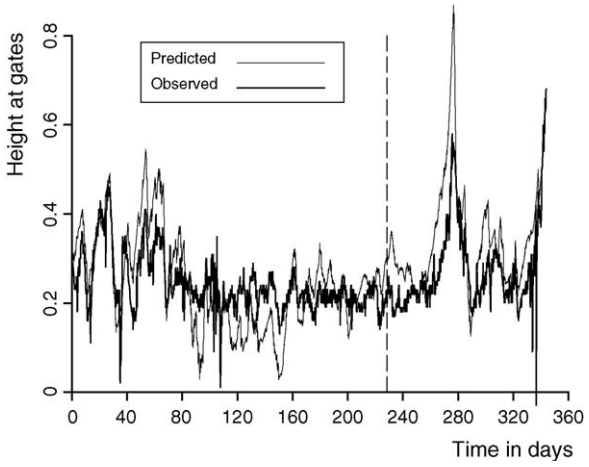


Fig. 2 – Observed water height in Ringkøbing Fjord over a year, along with heights predicted by the revised model.

3.3. *Population dynamics in the Ross Sea*

The Ross Sea in Antarctica has been the focus of many studies (Arrigo et al., 2003) because it has a relatively simple food web in comparison to open ocean systems. Bacteria, protists, and larger grazers play only a minor role, which means that we can safely ignore many processes, such as microbial interactions. Moreover, two recent scientific programs have collected field data from the southwestern Ross Sea over nearly complete growth cycles, which we can use to evaluate our approach to model revision. The most important organism in this ecosystem is phytoplankton, which undergoes repeated cycles of population increase and decrease, but measurements are also available for nitrate concentrations and sea ice coverage.

Incorporating domain knowledge from our team's biological oceanographer (Arrigo), we developed the initial process model shown in Table 1, which relates resources such as light and nutrients to phytoplankton growth. The model encodes much existing knowledge about how variables interact, but uncertainty about several components led us to consider alternative generic processes like those in Table 3. These include mechanisms for zooplankton grazing on phytoplankton, nitrate remineralization, and detritus loss. Because zooplankton was not measured, we treated it as an unobserved theoretical variable.

We have two sets of daily measurements of phytoplankton and nitrate concentrations in the Ross Sea, along with light levels and ice coverage, each spanning 188 days for two consequent years. We used the first year's observations as training data, combined with the generic processes from Table 3, to generate a revised model. In this run, we told RPM to

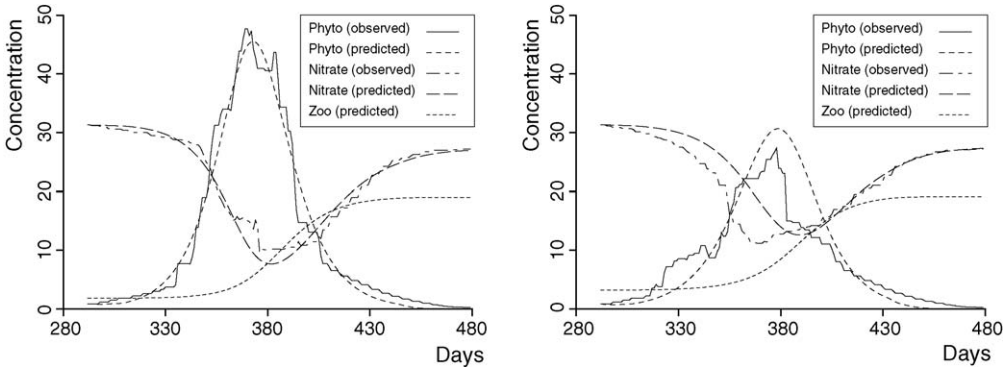


Fig. 3 – Observed phytoplankton and nitrate concentrations in the Ross Sea, along with predictions from the best revised process model, for training (left) and test (right) data.

consider removing from the initial model only the process phyto_uptakes_nitrate, but to improve the parameters of all processes and to consider adding instances of the generic processes grazing, Ivlev_rate, nutrient_remineralization, and detritus_loss_to_remin.

Detailed inspection of the results suggested that RPM's revisions were ecologically plausible. The best model at distance one added the process nutrient_remineralization, which involves restoring nitrate ions into the water from the detritus of dead phytoplankton. The best candidate at distance two also added the process detritus_loss_to_remin, which is a conservation term that balances mass transfer from detritus to nitrate ions. The next two changes introduced grazing and Ivlev_rate, which together describe the activity of zooplankton grazing on phytoplankton.

Fig. 3 shows the observed trajectories for the first year, along with trajectories generated by the best revised model, which has the four additions just described. The fit to both concentrations is quite good, with a mean squared error of 4.486 for phytoplankton and 2.010 for nitrate. The errors on the training set for the original model structure with revised parameters were 8.431 and 236.2, respectively, which indicates that the additional processes are helping substantially to fit the observations.

However, we want models that do more than match and explain such training data; we also want them to generalize well to unseen time series. When we used the best revised model to predict trajectories on the second year, we found that it produced almost exactly the same behavior, even though the observed peak for phytoplankton was much lower than in the first year. Inspection of the model suggested that ice differences across the years had little effect on phytoplankton growth, although this had originally seemed to us a likely explanation of differences between the two years.

But recall that initial values for unobservable variables are fit by RPM along with other parameters, and our test run assumed these were the same for the second year. It seemed plausible that differences in these values, especially for zooplankton, might account for the altered behavior. Thus, we ran the parameter estimation module on the revised model, letting it alter the initial values on the second year's data while retaining other parameters found on the first year.

Fig. 3 also presents the trajectories predicted by this slightly altered model, along with the concentrations observed during the second year. The height of the phytoplankton peak is much closer than the one predicted by the unmodified initial values. The mean squared error was 15.711 for phytoplankton and 9.377 for nitrate, as compared with 60.366 and 256.96 for the model structure from Table 1 with revised parameters. The timing of the nitrate trough occurs somewhat earlier than predicted, but the overall fit seems reasonable.

The figures also plot the inferred concentrations of zooplankton for both years, showing a higher initial level for the second year, which accounts for the lower peak because exponential growth of the zooplankton depletes the phytoplankton more rapidly. The graphs also reveal that the model does not predict a zooplankton decrease when food is no longer available. This would not alter the main effect we were seeking, but it does indicate that the model would be more plausible with another process, analogous to phyto_loss in Table 1, for zooplankton.

Informal analysis of the revised model's behavior also suggested another explanation of the lowered phytoplankton peak: the presence of a mechanism that causes phytoplankton to absorb more nitrate when it has insufficient light. In response, we added a new generic process that decreases nitrate as a linear function of the effective light but does not increase the concentration of phytoplankton. We ran RPM with this additional background knowledge on the first year's data and examined the results.

On this run, the best-scoring candidate on the training data remained the revised model we have already discussed, but the second best model included the new process that reduces nitrate concentration when light is not abundant. The mean squared errors were 4.639 for phytoplankton and 2.911 for nitrate on the first (training) year, whereas they were 21.039 and 9.414, respectively, for the second (test) year. Both the training and test errors are nearly as low as those for the best model. Fig. 4 presents the predicted trajectories for both years, which indicates that the qualitative fit is good and that the inferred levels for zooplantkon are closer in the two curves, so that the explanatory power comes from the new nitrate-using process.

An alternative, but roughly equivalent, interpretation is that the nitrogen-to-carbon ratio for phytoplankton varies as

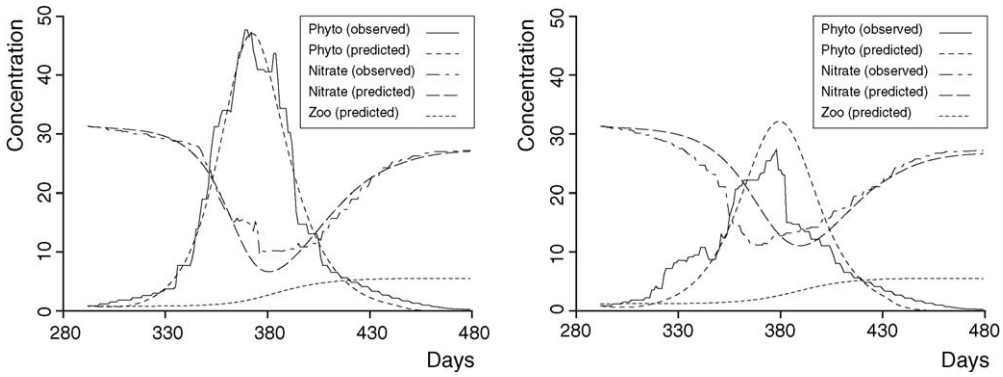


Fig. 4 – Observed phytoplankton and nitrate concentrations in the Ross Sea, along with predictions from the second-best revised model, for training (left) and test (right) data.

a function of light availability. This insight is an important one from the perspective of ecological modeling, in that previous accounts have assumed phytoplankton's nitrogen quota is constant. If the revised process model is correct in claiming that reduced light levels (caused by heavier ice cover) increase nitrogen requirements,[2] then this suggests we should reexamine models for a broad class of aquatic ecosystems that operate under similar environmental conditions.

3.4. *Additional analyses of model revision*

Earlier in the paper, we argued that revising a process model had advantages over inducing a similar model from scratch. We were especially interested in how model quality varied with distance from the initial candidate, which we measured as the symmetric structural difference between the processes in the initial model and revised one.

Fig. 5 plots the training and test error for Ringkøbing Fjord against the distance from the initial model for the revisions suggested by RPM. The error decreases initially as one moves away from the initial model, but after a distance of two the errors increase again. This is not surprising if the 'true' model falls at an intermediate distance from the initial one. More interesting is that the curves for training and test set error are similar, which suggests our revision method does not overfit the training data in this domain, despite the fact that RPM includes no explicit features to guard against this danger.

For comparison purposes, we also ran IPM on the same set of processes and training data for Ringkøbing Fjord. The IPM algorithm took about six times longer than RPM to finish investigating its model space, which consisted of 217 different model structures. The best-scoring model found by IPM had mean squared error on the test data (0.0081) that was slightly lower than the best induced by RPM (0.0099), but it fared much worse in terms of qualitative prediction. In fact, this model did not track any of the major variations in water height because it predicted a nearly flat trajectory over time.

Analogous distance curves for the Ross Sea data sets also appear in Fig. 5. In this case, the minimum error for both curves occurs with four structural revisions, although the test curve does not follow the systematic U shape of the training curve. In addition, we trained IPM on the first year's data and tested its results on the second year. The program took about 20 hours to explore 121 different model structures, considerably more time than RPM, which took 5 hours to consider 40 candidate structures. The best-scoring models produced by the two systems had very similar structures and comparable error rates on the training and test years.

These results provide evidence that, by revising an initial model instead of learning from scratch, RPM can use relatively little search to find alternatives. These altered models improve greatly on the original model structure in terms of predictive accuracy while remaining consistent with domain knowledge, and the system finds them in a fraction of the time required to construct a model entirely from generic processes.

4. Related and future work

Computational methods for model revision are certainly not a new idea. Early research (e.g., Towell, 1991) focused on supervised learning for classification tasks, but supported modification of models with theoretical terms and offered a general framework from which we have borrowed. For instance, Ourston and Mooney's (1990) EITHER utilized search operators for adding and deleting rules, which correspond to our operators for adding and removing processes. They also included ones for adding and removing conditions on rules, which are quite different from our scheme for parameter revision. Still, their view of model revision as searching from an initial model, guided by fit to training data, is a general one that is relevant to revision of quantitative process models.

A different tradition has explored methods for revising qualitative causal models of scientific phenomena. Early examples included Rajamoney's (1990) COAST system, which used ideas from qualitative physics to improve models of fluid and heat flow, and Kulkarni and Simon's (1990) KEKADA, which reproduced many steps in Krebs' discovery of the urea cycle. These systems altered their models incrementally, but later work has utilized nonincremental methods with simpler representations of causal connections. Recent examples include Bay et al.'s (2003) method for revising qualitative models of gene regulation, which carries out greedy search guided by

[2] Needoba and Harrison (2003) present more direct evidence that such a mechanism exists, which lends credibility to our proposal.

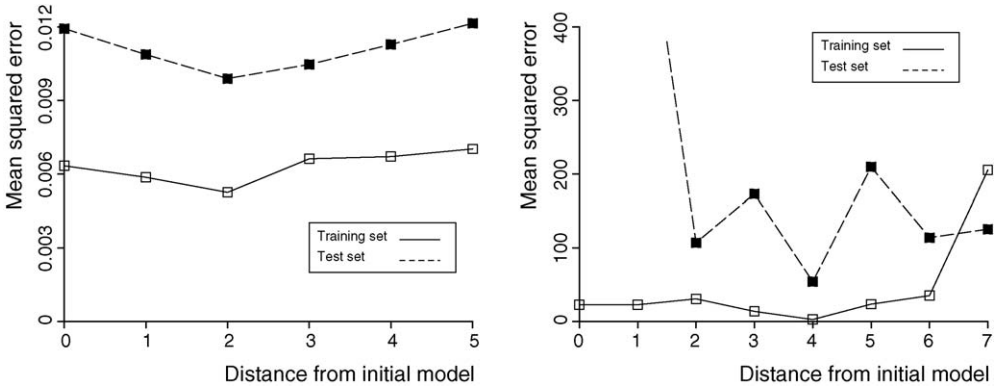


Fig. 5 – Training and testing errors for best RPM models for Ringkøbing Fjord (left) and the Ross Sea (right) as a function of distance from initial model.

candidate models' fits to the data. Bryant et al. (2001) report a different approach that uses abductive logic programming to extend a qualitative model of metabolic control.

A more closely related line of research has addressed the revision of quantitative models of ecosystem behavior. Chown and Dietterich (2000) report a system that improves the parameters in a complex ecosystem model by decomposing it into more tractable subproblems. Both Saito et al. (2001) and Todorovski et al. (2003) describe methods that revise the parameters and functional forms in a nondynamic ecosystem model. A key difference is that our approach introduces the notion of processes, which provides a useful framework for encoding domain knowledge that constrains search and produces more interpretable results. However, Whigham and Recknagel (2001) have also explored methods for inducing ecosystem models cast in process terms, as has Todorovski (2003) in his recent work on dynamical systems.

Although most research has emphasized automated revision methods, a few groups have developed interactive systems. For instance, Mitchell et al.'s (1997) program encourages metallurgists to actively direct its search for quantitative relations, whereas Mahidadia and Compton (2001) report an environment for the interactive revision of qualitative causal models in neuroendocrinology. We believe that most scientists will prefer computational discovery tools that keep them involved in the revision effort, and we intend to embed future versions of RPM in such an interactive environment for scientific model development.

Naturally, we also intend to extend our revision methods on various fronts. One drawback of the current system is that parameter estimation takes 99.9% of the computation time, which limits the number of models with different structure it can consider effectively. Preliminary studies with a hierarchical method for multiple shooting (Horbelt et al., 2001) have shown promise for speeding this component. More rapid techniques for parameter fitting should also let us augment RPM to consider new conditions on processes during the revision effort, thus increasing its flexibility.

Although we found little evidence for overfitting in our three domains, we also plan to investigate several methods for mitigating this problem when it does occur. These include averaging parameter estimates with statistical resampling techniques and incorporating uncertainty explicitly in the estimated parameters used for simulation. We should also introduce guards against overfitting in the search for model structures. Some variation on minimum description length is an obvious choice, but this should incorporate a bias toward candidates with structures similar to the initial model. Saito et al. (2002) such a "minimal change principle" in their work on revision of qualitative biological models, whereas Todorovski et al. (2003) have proposed a similar idea in the context of revising quantitative ecological models.

Finally, RPM's current reliance on exhaustive search lets it handle only relatively small model spaces. Future versions should replace this approach with a heuristic method that scales to more complex models and to more extensive changes. The generation of model structures should be more closely linked to parameter estimation, thus making both more efficient. We should also extend the framework to support revision of models with subsystems, which ecosystem modelers often utilize when dealing with complex domains. This would reduce search by drawing on knowledge about likely subsystems rather than isolated processes, and thus further improve scaling ability.

5. Concluding remarks

In this paper, we reported an approach to computational scientific modeling that focuses on the revision of existing models rather than on their construction. Unlike earlier work in this area, we utilized the formalism of quantitative process models, which support explanatory accounts of continuous time series in terms of unobservable variables and processes. Our specific algorithm, RPM, lets users specify an initial model, a data set, and a set of allowed revisions. The latter can include specific processes that may be deleted, specific processes for which the parameters may be altered, and generic processes that may be added. We demonstrated this system's abilities in three environmental domains, one involving changes in water height in a Danish Fjord and another concerning population dynamics in the Ross Sea.

Our experimental results were generally encouraging. In each domain, we showed that RPM found meaningful

revisions that had substantially lower error than the original model structure. Moreover, we found that these models were generally as accurate as those produced by IPM, which composes models entirely from generic processes, but were obtained with less search and in less time. However, we also identified some limitations of the system that future work should address. In summary, our approach builds on previous research in model revision and scientific discovery, but extends their ideas in ways that are useful for fields like environmental science, which often utilize quantitative models of dynamical systems.

Acknowledgements

This research was supported by NSF Grant no. IIS-0326059 and by NTT Communication Science Laboratories, Nippon Telegraph and Telephone Corporation. We thank Ljupčo Todorovski for providing the initial models and data for the Ringkøbing Fjord domain and Jed Crosby for his analysis of that domain. We also thank Kazumi Saito and Dileep George for their initial work on the predator–prey ecosystem.

REFERENCES

Arrigo, K.R., Worthen, D.L., Robinson, D.H., 2003. A coupled ocean-ecosystem model of the Ross Sea: 2. Iron regulation of phytoplankton taxonomic variability and primary production. J. Geophys. Res. 108 (C7), 3231, 10.1029/2001JC000856.

Bay, S.D., Shrager, J., Pohorille, A., Langley, P., 2003. Revising regulatory networks: from expression data to linear causal models. J. Biomed. Inform. 35, 289–297.

Bryant, C.H., Muggleton, S.H., Oliver, S.G., Kell, D.B., Reiser, P., King, R.D., 2001. Combining inductive logic programming, active learning and robotics to discover the function of genes. Electron. Trans. Artif. Intell. 5-B1 (012), 1–36.

Chown, E., Dietterich, T.G., 2000. A divide and conquer approach to learning from prior knowledge. Proceedings of the 17th International Conference on Machine Learning, Morgan Kaufmann, San Francisco, pp. 143–150.

Forbus, K.D., 1984. Qualitative process theory. Artif. Intell. 24, 85–168.

Horbelt, W., Voss, H.U., Timmer, J., 2001. Parameter estimation in nonlinear delayed feedback systems from noisy data. Phys. Lett. A 299, 513–521.

Jost, C., Adiriti, R., 2000. Identifying predator–prey processes from time-series. Theoret. Popul. Biol. 57, 325–337.

Kulkarni, D., Simon, H.A., 1990. Experimentation in machine discovery. In: Shrager, J., Langley, P. (Eds.), Computational Models of Scientific Discovery and Theory Formation. Morgan Kaufmann, San Mateo, CA.

Langley, P., 2000. The computational support of scientific discovery. Int. J. Human-Comput. Stud. 53, 393–410.

Langley, P., George, D., Bay, S., Saito, K., 2003. Robust induction of process models from time-series data. Proceedings of the 20th International Conference on Machine Learning, AAAI Press, Washington, DC, pp. 432–439.

Levenberg, K., 1944. A method for the solution of certain problems in least squares. Q. Appl. Math. 2, 164–168.

Mahidadia, A., Compton, P., 2001. Assisting model discovery in neuroendocrinology. Proceedings of the Fourth International Conference on Discovery Science, Springer, Washington, DC, pp. 214–227.

Marquardt, D., 1963. An algorithm for least-squares estimation of nonlinear parameters. SIAM J. Appl. Math. 11, 431–441.

Mitchell, F., Sleeman, D., Duffy, J.A., Ingram, M.D., Young, R.W., 1997. Optical basicity of metallurgical slags: a new computer-based system for data visualisation and analysis. Ironmak. Steelmak. 24, 306–320.

Needoba, J.A., Harrison, P.J., 2003. Influence of low light and a light: dark cycle on NO_3^- uptake, intracellular NO_3^-, and nitrogen isotope fractionation by marine phytoplankton. J. Phycol. 40, 505–516.

Ourston, D., Mooney, R., 1990. Changing the rules: a comprehensive approach to theory refinement. Proceedings of the Eighth National Conference on Artificial Intelligence, AAAI Press, Boston, pp. 815–820.

Rajamoney, S., 1990. A computational approach to theory revision. In: Shrager, J., Langley, P. (Eds.), Computational Models of Scientific Discovery and Theory Formation. Morgan Kaufmann, San Mateo, CA.

Saito, K., Bay, S., Langley, P., 2002. Revising qualitative models of gene regulation. Proceedings of the Fifth International Conference on Discovery Science, Springer, Lubeck, Germany, pp. 59–70.

Saito, K., Langley, P., Grenager, T., Potter, C., Torregrosa, A., Klooster, S.A., 2001. Computational revision of quantitative scientific models. Proceedings of the Fourth International Conference on Discovery Science, Springer, Washington, DC, pp. 336–349.

Todorovski, L., 2003. Using domain knowledge for automated modeling of dynamic systems with equation discovery. Doctoral Dissertation. Faculty of Computer and Information Science, University of Ljubljana, Slovenia.

Todorovski, L., Dzeroski, S., Langley, P., Potter, C., 2003. Using equation discovery to revise an earth ecosystem model of carbon net production. Eco. Model. 170, 141–154.

Towell, G., 1991. Symbolic knowledge and neural networks: insertion, refinement, and extraction. Doctoral Dissertation. Computer Sciences Department, University of Wisconsin, Madison.

Whigham, P.A., Recknagel, F., 2001. Predicting Chlorophyll-*a* in freshwater lakes by hybridising process-based models and genetic algorithms. Ecol. Model. 146, 243–251.