ELSEVIER

Ecological Modelling 113 (1998) 71–81

ECOLOGICAL MODELLING

Modelling and prediction of phytoplankton growth with equation discovery

Ljupčo Todorovski [a,*], Sašo Džeroski [b], Boris Kompare [c]

[a] *University of Ljubljana, Faculty of Medicine, Vrazov trg 2, 1105 Ljubljana, Slovenia*
[b] *Jožef Stefan Institute, Jamova 39, 1111 Ljubljana, Slovenia*
[c] *University of Ljubljana, Faculty of Civil and Geodetic Engineering, Hajdrihova 28, 1001 Ljubljana, Slovenia*

Abstract

In contrast with traditional modelling methods, which are used to identify parameter values of a model with known structure, equation discovery systems identify the structure of the model also. The model generated with such systems can give experts a better insight into the measured data and can be also used for predicting future values of the measured variables. This paper presents LAGRAMGE, an equation discovery system that allows the user to define the space of possible model structures and to make use of domain specific expert knowledge in the form of function definitions. We use LAGRAMGE to automate the modelling of phytoplankton growth in lake Glumsoe, Denmark. The structure of the model constructed with LAGRAMGE agrees with human experts' expectations. The model can be successfully used for long term prediction of phytoplankton concentration during algal blooms.

Keywords: Artificial intelligence; Automated modelling; Equation discovery; LAGRAMGE; Phytoplankton; System identification

1. Introduction

The task of modelling dynamic systems is to find a model that describes an observed behavior. A model of a dynamic system is usually a set of differential equations that specifies the change of system variables over time. Mainstream system identification methods, surveyed in Ljung (1993) work under the assumption that the model structure, i.e. the form of the differential equations, is known. The task is then to determine the values of the constant parameters in the equations, so that the model fits measured data. The structure of the equations is provided by the human expert and is based on the theoretical knowledge about the domain at hand. The formalization of the theoretical knowledge in the domains of use where no strict mathematical laws are available can be problematic. In such domains, models that are linear in the system variables are used (Ljung, 1993). However, these models typically do not give sufficient insight into the physical back-

* Corresponding author. Tel.: +386 61 313233; fax: +386 61 311540; e-mail: ljupco.todorovski@mf.uni-lj.si

0304-3800/98/$ - see front matter
PII S0304-3800(98)00135-5

ground of the underlying dynamic system and their accuracy depends only on the precision and frequency of the available measurements.

Equation discovery systems, such as LAGRANGE (Džeroski and Todorovski, 1993) and GOLDHORN (Križman et al., 1995), do not assume a prescribed model structure, but rather explore a space of (possibly non-linear) equations. They help human experts to identify the structure of the model as well as the values of the constant parameters. Equation discovery systems can be used for automated modelling of ecological dynamic systems. Kompare (Kompare, 1995) used LAGRANGE and GOLDHORN to produce a model for predicting algal growth in the Lagoon of Venice. Several problems arise when using these systems for modelling experimental data. LAGRANGE discovered some equations predicting the optimal temperature for algal growth, however, no good equations were discovered, from the viewpoint of what human experts expected. This was due to the high level of noise in the data. Methods for discovery from noisy data are incorporated in GOLDHORN, so reasonable equations were discovered, however, many equations with unacceptable structure were ranked as better fitted.

These problems led to the idea of restricting the space of possible equations considered in the process of discovery by taking into account the expert's knowledge of the domain at hand. In the area of machine learning, declarative language bias (Dehaspe and DeRaedt, 1995) is used to specify the hypothesis space. In the task of equation discovery, this would be the space of all possible equations, or more precisely, the space of all possible equation structures. It has been observed that smaller hypothesis spaces lead to improved performance of the learned concept (model) on a test set of unseen cases (Nédellec et al., 1996).

In this paper, we present the equation discovery system LAGRAMGE[1] (Todorovski and Džeroski, 1997) that uses context free grammars as a formalism for specifying the form of discovered equations. The grammar can use the usual mathematical operators defined in the C programming language, as well as additional functions defined by the grammar at hand. The grammar is specified according to domain specific knowledge, and focuses the equation discovery process on equations with structure that is acceptable and comprehensible within the domain of use. The context free grammar does not necessarily specify the precise structure of the model as in mainstream system identification methods, but can only be used to indicate the form of the expressions on the right-hand side of the equations.

LAGRAMGE was applied to the problem of modelling phytoplankton growth in Lake Glumsoe, Denmark. The modelling was conducted on the basis of only 14 measurements over a period of 2 months. Using the background knowledge of human experts regarding the dynamic behavior of the system variables between measurement points, several data sets suitable for equation discovery were created. Expert knowledge was also used for building the context free grammar used in LAGRAMGE. The structure of the equations discovered by LAGRAMGE makes sense from an ecological point of view. The equations can also be used as accurate predictors for phytoplankton growth.

2. The equation discovery system LAGRAMGE

2.1. Problem definition

The problem of equation discovery, as addressed by LAGRAMGE, can be defined as follows. Given are:

- a context free grammar $G = (N, T, P, S)$ (Section 2.2) and
- input data $D = (V, v_d, M)$, where
  $V = \{v_1, v_2, \ldots v_n\}$ is a set of domain variables,
  $v_d \in V$ is the dependent variable and
  M is a set of one or more measurements. Each measurement is a table of measured values of the domain variables at successive time points (Table 1).

[1] This is a deliberate misspelling of the name of the equation discovery system LAGRANGE, the predecessor of LAGRAMGE. The letter N is replaced with M, so that the second part of the acronym reads Gram as in grammar. Namely, declarative bias based on grammars is used in LAGRAMGE.

Find an equation for expressing the dependent variable v_d in terms of variables in V. This equation is expected to minimize the discrepancy between the measured and calculated values of the dependent variable. The equation can be:

- differential, i.e. of the form $dv_d/dt = \dot{v}_d = E$, or
- ordinary, i.e. of the form $v_d = E$,

where E is an expression that can be derived from the context free grammar G.

2.2. The declarative bias formalism

The syntax of the expressions on the right hand side of the equation is prescribed with a context free grammar (Hopcroft and Ullman, 1979). A context free grammar contains a finite set of variables (also called nonterminals or syntactic categories), each of which represents expressions or phrases in a language (in equation discovery, nonterminals represent sets of expressions that can appear in the equations). The expressions represented by the nonterminals are described in terms of nonterminals and primitive symbols called terminals. The rules relating the nonterminals among themselves and to terminals are called productions.

The original motivation for the development of context free grammars was the description of natural languages, e.g. a simple grammar for deriving sentences consists of the productions *sentence* → *noun verb*, *noun* → *phytoplankton*, *noun* → *zooplankton*, and *verb* → *grows*. Here, *sentence*, *noun* and *verb* are nonterminals, while words that actually appear in sentences (i.e. *phytoplankton*, *grows*) are terminals. The sentences *phytoplankton grows* and *zooplankton grows* can be derived with this grammar.

Table 1
Table of measured values of the domain variables at successive time points

Time	v_1	v_2		v_n
t_0	$v_{1,0}$	$v_{2,0}$	…	$v_{n,0}$
t_1	$v_{1,1}$	$v_{2,1}$	…	$v_{n,1}$
t_2	$v_{1,2}$	$v_{2,2}$	…	$v_{n,2}$
t_m	$v_{1,m}$	$v_{2,m}$	…	$v_{n,m}$

We denote a context free grammar as a tuple $G = (N, T, P, S)$, where N and T are finite disjoint sets of nonterminals and terminals, respectively. P is a finite set of productions; each production is of the form $A \rightarrow \alpha$, where A is a nonterminal and α is a string of symbols from $N \cup T$. We use the notation $A \rightarrow \alpha_1 | \alpha_2 | \ldots | \alpha_k$ for a set of productions for the nonterminal A: $A \rightarrow \alpha_1$, $A \rightarrow \alpha_2$, …, $A \rightarrow \alpha_k$. Finally, S is a special nonterminal called starting symbol.

Grammars used to describe declarative biases for equation discovery have several symbols with special meanings. The terminal const$\in T$ is used to denote a constant parameter in an equation that has to be fitted to the input data. The terminals v_i are used to denote variables from the input domain D. Finally, the nonterminal $v \in N$ denotes any variable from the input domain. Productions connecting this nonterminal symbol to the terminals v_i are attached to v automatically, i.e. $\forall v_i \in V: v \rightarrow v_i \in P$.

The only restriction on the grammar G is that the right sides of the productions in P have to be expressions that are legal in the C programming language. This means that we can use all C built-in operators and functions in the grammar. Additional functions, representing background knowledge about the domain at hand, can be used, as long as they are defined in conjunction with the grammar. Note that the derived equations may be non-linear in both the constant parameters and the system variables.

Expressions can be derived by grammar G from the nonterminal symbol S by applying productions from P. We start with the string w consisting of S only. At each step, we replace the leftmost nonterminal symbol A in string w with α, according to some production $A \rightarrow \alpha$ from P. When w consists solely of terminal symbols, the derivation process is over.

2.3. An example of an aquatic ecosystem

We illustrate the use of grammars in a simple aquatic ecosystem domain. A system of differential equations describes the evolution of the con-

Table 2
Context free grammar used as declarative bias for equation discovery in the simple aquatic ecosystem domain

```
double monod(double c, double v)
        {return(v / (v + c));}
```

$$
\begin{aligned}
N &= \{E, F, Y, v\} \\
T &= \{+, const, \cdot, \texttt{monod}, (, ,,), nut, phyt, zoo\} \\
P &= \left\{ \begin{array}{lcl} E & \rightarrow & const \mid const \cdot F \mid E + const \cdot F \\ F & \rightarrow & v \mid Y \mid v \cdot Y \\ Y & \rightarrow & \texttt{monod}(const, v) \end{array} \right\} \\
S &= E
\end{aligned}
$$

centrations of nutrient (nut), phytoplankton (phyt) and zooplankton (zoo) in an aquatic environment (Crispi and Mosetti, 1993):

$$\dot{\text{nut}} = \frac{\text{nut} \cdot \text{phyt}}{k_N + \text{nut}}$$

$$\dot{\text{phyt}} = \frac{\text{nut} \cdot \text{phyt}}{k_N + \text{nut}} - r_P \cdot \text{phyt} - \frac{\text{phyt} \cdot \text{zoo}}{k_P + \text{phyt}}$$

$$\dot{\text{zoo}} = \frac{\text{phyt} \cdot \text{zoo}}{k_P + \text{phyt}} - r_Z \cdot \text{zoo}.$$

A sample grammar that can be used as declarative bias in the aquatic ecosystem domain, is given in Table 2. The productions $v \rightarrow$ nut|phyt|zoo connecting the nonterminal v with the domain variables are automatically added to the grammar. The definition of the background knowledge function `monod` in the C programming language is also attached to the grammar.

The definition of the function `monod` is based on expert background knowledge in population dynamics (Monod, 1942; Jørgensen, 1986; Crispi and Mosetti, 1993; Bendoricchio et al., 1994). This function is used to describe population growth where v is a nutrient variable and c a saturation constant. The `monod` function is incorporated in expressions through the production $Y \rightarrow$ `monod` (const, v). Productions for the nonterminal F can be used to combine domain variables and `monod` terms in multiplicative terms. Finally, linear combinations of these multiplicative terms are constructed with productions for the starting nonterminal E. Note that the derived expressions are not necessarily linear in either the constant parameters or the system variables, because of the presence of a constant parameter and a variable in the denominator of the `monod` term.

We can derive the expression const · phyt · nut/(const + nut) from the grammar with the derivation in Table 3. We use the production $E \rightarrow const \cdot F$ at the beginning, because the desired expression consists of one single multiplicative term. To derive an expression with more than one multiplicative term, we can start with the production $E \rightarrow E +$ const · F and use it repeatedly, until the desired number of terms is derived.

2.4. LAGRAMGE—the algorithm

Expressions generated by the context free grammar G contain one or more special terminal symbols const. A nonlinear fitting method is applied to determine the values of these parameters. The fitting method minimizes the value of the error function Error($\boldsymbol{c}$), i.e. if $\boldsymbol{c}$ is the vector of constant parameters in expression E, then the result of the fitting algorithm is a vector of parameter values

Table 3
Derivation of the expression const · phyt · `monod` (const, nut)

Expression	Production used
$E \rightarrow$	$E \rightarrow$ const · F
const · $F \rightarrow$	$F \rightarrow v \cdot Y$
const · $v \cdot Y \rightarrow$	$v \rightarrow$ phyt
const · phyt · $Y \rightarrow$	$Y \rightarrow$ `monod` (const, v)
const · phyt · `monod` (const, v)	$v \rightarrow$ nut
const · phyt · `monod` (const, nut)	

$\boldsymbol{c}^*$, such that $\text{Error}(\boldsymbol{c}^*) = \min_{\boldsymbol{c}\in R^{n_c}}\{\text{Error}(\boldsymbol{c})\}$. The error function, is a sum of squared errors function, defined in the following manner:

- for a differential equation of the form $\partial v_\text{d}/\partial t = E$:$\text{Error}(\boldsymbol{c})$

$$= \Sigma_{i=0}^{m}\left[v_{\text{d},i} - \left(v_{\text{d},0} + \int_{t_0}^{t_i} E(\boldsymbol{c}, v_1, \ldots v_n)\right)\right]^2, \text{ and}$$

- for an ordinary equation of the form $v_\text{d} = E$:$\text{Error}(\boldsymbol{c})$

$$= \Sigma_{i=0}^{m}(v_{\text{d},i} - E(\boldsymbol{c}, v_{1,i}, \ldots v_{\text{d}-1,i}, v_{d+1,i}, \ldots v_{n,i}))^2,$$

where m is the size of the measurement table and $v_{j,i}$ the value of the system variable v_j at time t_i. Note that in the case of calculating the error function for differential equations, we use the integral of the expression on the right hand side of the equation instead of the derivative of the dependent variable. This is because the error of algorithms for numerical integration is in general smaller than the error involved of numerical derivation. We use a simple trapezoid formula for numerical integration with the same step size as the time step between successive measurements in the measurement table. The downhill simplex and Levenberg–Marquardt algorithms (Press et al., 1986) can be used to minimize the error function.

Furthermore, the value of a heuristic function for the expression is evaluated. It is equal to the sum of squared errors value SSE calculated by the fitting method ($\text{SSE}(E) = \text{Error}(\boldsymbol{c}^*)$). An alternative heuristic function MDL (minimal description length) can be used, that takes into account the length l of expression E:

$$\text{MDL}(E) = \text{SSE}(E) + \frac{l}{10 \cdot l_{\max}}\sigma_{v_\text{d}},$$

where $l_{\max}$ is the length of the largest expression generated by the grammar and σ_{v_d} is the standard deviation of the dependent variable v_d. The length is measured as the number of terminals in the expression. The MDL heuristic function prefers shorter equations.

A context free grammar can in principle derive an infinite number of expressions (equations). LAGRAMGE thus uses a bound on the complexity (depth) of the derivation used to produce the equation (Todorovski and Džeroski, 1997). The LAGRAMGE algorithm exhaustively or heuristically searches for the best equation (according to the selected heuristic function) within the allowed complexity (depth) limits.

3. Lake Glumsoe

Lake Glumsoe (Jørgensen et al., 1986) is situated in a sub-glacial valley in Denmark. It is shallow with an average depth of ~ 2 m and its surface area is 266000 m^2. For several years, it was receiving mechanically–biologically treated waste water from a community with 3000 inhabitants and a surrounding area which was mainly agricultural with almost no industry. The high nitrogen and phosphorus concentration in the treated waste water has caused hypereutrophication. The lake contained no submerged vegetation, probably due to the low transparency of the water and oxygen deficit at the bottom of the lake.

Concentrations of phytoplankton (phyt), zooplankton (zoo), soluble nitrogen (nitro) and soluble phosphorus (phosp) were considered relevant for modelling the phytoplankton growth. State variables were measured at 14 distinct time points, over a period of 2 months. The amount of measured data itself was far too small for equation discovery, so additional processing was applied to obtain a dataset suitable for equation discovery (Kompare, 1995; Demšar, 1996). Firstly, dotted graphs of the measurements were plotted and given to three human experts to draw a curve that, in their own opinion, described the dynamic behavior of the observed system variable between the measured points. A properly plotted expert curve can be regarded as an additional source of reliable data. Curves drawn by the human experts were then smoothed with Besier splines. Finally, three data sets were obtained by sampling the splines derived from each of the three human expert' approximations at regular time intervals with time step $h = 0.1$ day. The dynamic behavior of the phytoplankton as represented by each of the three data sets is shown in Fig. 1.

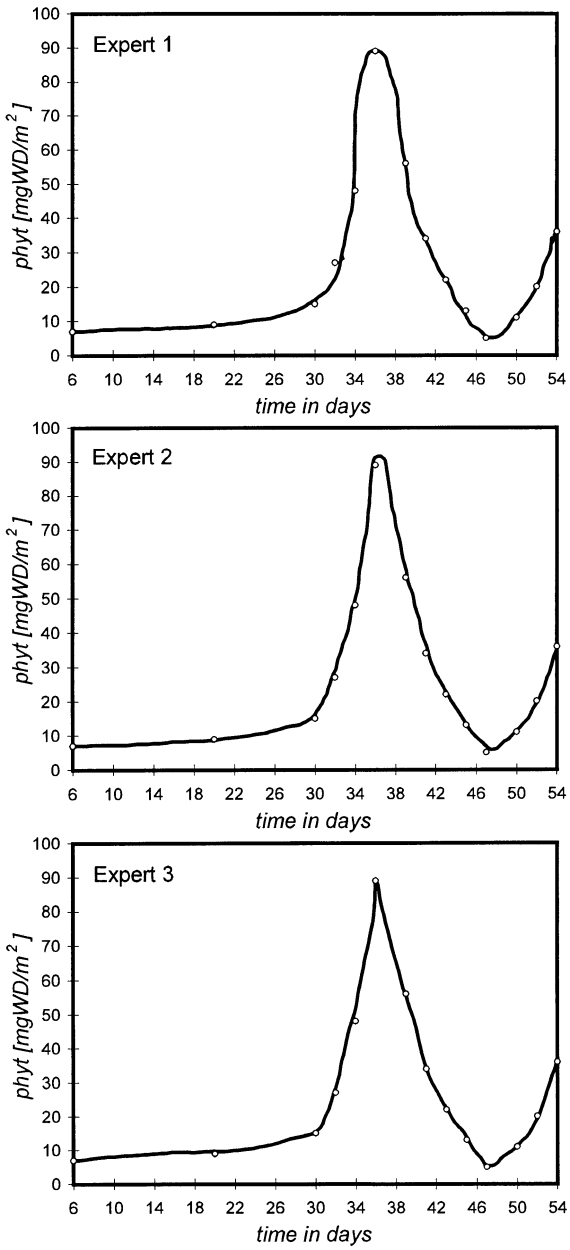

Fig. 1. Phytoplankton growth as seen by three domain experts.

4. Experiments

The grammar given in Table 4 was used in the experiments. It was constructed by taking into account ecological background knowledge on algal growth (Monod, 1942; Jørgensen, 1986; Crispi and Mosetti, 1993; Bendoricchio et al., 1994). Phosphorus and nitrogen are nutrients for phytoplankton and can thus appear in `monod` terms (productions for nonterminals Y and v_Y). Other terms describe the decay of phytoplankton ($-\text{const} \cdot \text{phyt}$) and the feeding of zooplankton on phytoplankton ($-\text{const} \cdot \text{phyt} \cdot \text{zoo}$). At the maximum derivation depth 4 used in our experiments, 72 equations can be derived from the grammar. The values of the constant parameters in the equations specified by the grammar are constrained to be positive.

In the first set of experiments, we used the 'leave one out' testing method: LAGRAMGE was given two sets of data for equation discovery, and the best equation discovered was then tested on the remaining data set. The equation was tested on the task of predicting phytoplankton growth.

In experiments with the MDL heuristic function (all possible 72 equations were considered), the best equation discovered by LAGRAMGE was chosen that satisfied the constraints for the parameters' values. The three equations obtained had the same structure:

$$\text{phyt} = \text{const}_1 \cdot \text{phyt} \cdot \frac{\text{phosp}}{\text{const}_2 + \text{phosp}} - \text{const}_3 \cdot \text{phyt}$$

The structure of the equations discovered makes sense from an ecological point of view. It tells us that phosphorus is a limiting factor for phytoplankton growth in the lake.

We used the obtained equations for predicting the phytoplankton concentration in the lake on the testing set and then calculated the correlation coefficient between the measured and predicted values. The constant parameter values, as well as calculated correlation coefficients are shown in Table 5. It can be seen that all equations give accurate short term predictions for phytoplankton growth. Note, however, the differences in the values of the equation coefficients, which indicate that experts approximated the dynamics of phytoplankton growth in quite different ways. Furthermore, we tested the robustness of the predictor on increasing the prediction period. The summary of the results (correlation coefficients between mea-

Table 4
Context free grammar used for the Lake Glumsoe domain

```
double monod(double c, double v) {
        return(v / (v + c));
}
```

$$
\begin{aligned}
E &\rightarrow F \cdot phyt - const \cdot phyt \mid F \cdot phyt - const \cdot phyt - const \cdot phyt \cdot zoo \\
F &\rightarrow const \cdot Y \cdot Y \mid const \cdot Y + const \cdot Y \mid const \cdot Y \\
Y &\rightarrow \mathrm{monod}(const, v_Y) \mid v_Y \\
v_Y &\rightarrow phosp \mid nitro
\end{aligned}
$$

sured and predicted values) for prediction periods of 1, 2 and 5 days are given in Table 6.

Finally, we compared the accuracy of the obtained predictor with the accuracies of two simple predictors: no-change and same-change. The no-change predictor predicts that the value of the variable at the next time point will be the same as the present value ($\widehat{\mathrm{phyt}}(t+h) = \mathrm{phyt}(t)$). The same-change predictor predicts the same change of the value of the variable, as the change in previous time step ($\widehat{\mathrm{phyt}}(t+h) - \mathrm{phyt}(t) = \mathrm{phyt}(t) - \mathrm{phyt}(t-h)$). The graphs in Fig. 2 show the dependence of correlation coefficients between the measured values and values predicted by the three different predictors for increasing prediction period for all data sets.

The graphs show that the accuracy of the predictions decreases as the prediction time increases, which could be expected. The performance and robustness of all predictors are comparable. The same-change predictor has better performance than the one obtained with LAGRAMGE, especially on the third data set, but the LAGRAMGE predictor is more robust, i.e. has smaller oscillations of performance. The no-change predictor has the lowest accuracy on all data sets.

Table 5
Constant parameters' values and correlation coefficients for equations discovered by LAGRAMGE

Training data sets	$const_1$	$const_2$	$const_3$	r
1, 2	0.617	0.101	0.442	0.9994
1, 3	0.763	0.0797	0.592	0.9989
2, 3	0.383	0.444	0.155	0.9996

Recently, a new data set was obtained by applying a more sophisticated smoothing method to the graph plotted by the first human expert (Fig. 3). The new data set also includes the measurements of the temperature of the water in the lake (temp). Due to the fact that the second bloom might not be described by the same model, the last portion of the data was not taken into account.

The grammar used in the experiments with the new data set was the same as the one used in previous experiments (Table 4), except that the new production $v_Y \rightarrow \mathrm{temp}$ was added to allow the use of temperature in the `monod` terms. The best equation discovered by LAGRAMGE that satisfies the constraint for the constant parameters' values was:

$$
\dot{\mathrm{phyt}} = 0.553 \cdot \mathrm{temp} \cdot \mathrm{phyt} \cdot \frac{\mathrm{phosp}}{0.0264 + \mathrm{phosp}} - 4.35 \cdot \mathrm{phyt} - 8.67 \cdot \mathrm{phyt} \cdot \mathrm{zoo}
$$

Note that the structure of the equation discovered is similar to the structure of equations discovered on three data sets. It tells us that phosphorus and water temperature are the limiting factors for phytoplankton growth in the lake.

Table 6
Correlation coefficients between the actual phytoplankton concentration and the concentration predicted by the discovered equations for different prediction time periods

Training data sets	1 day	2 days	5 days
1, 2	0.9836	0.9413	0.7243
1, 3	0.9849	0.9552	0.8137
2, 3	0.9853	0.9566	0.7267

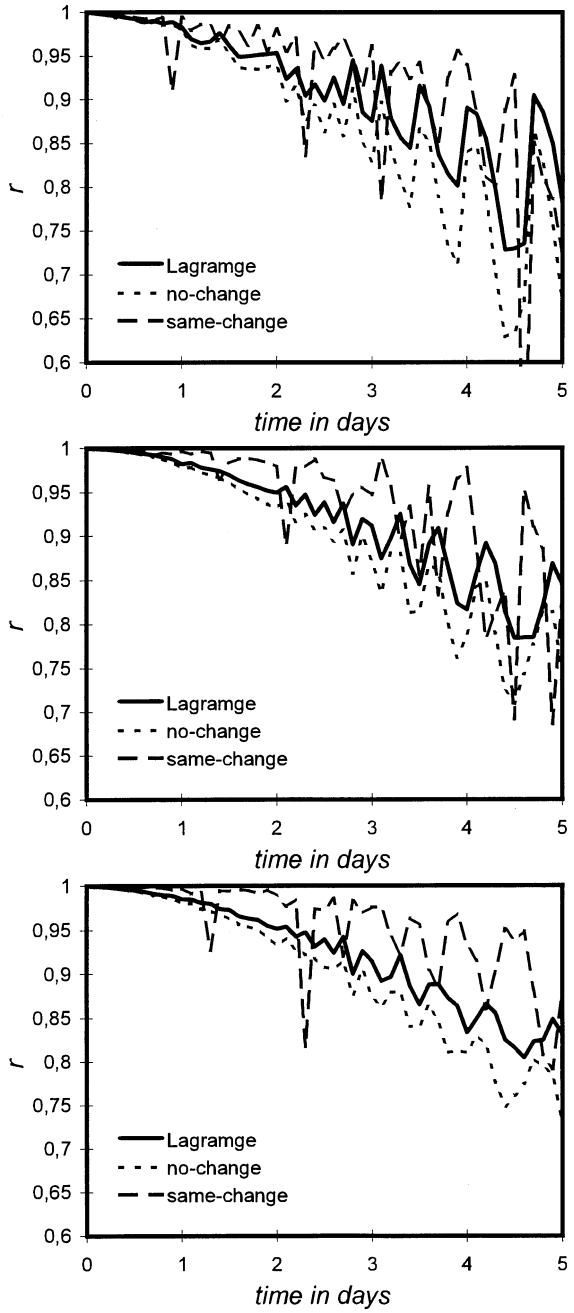

Fig. 2. Dependence of correlation coefficients between the measured values and values predicted with the three different predictors on increasing prediction period for all experimental data sets.

Next, a context free grammar for linear equations was used for equation discovery from the

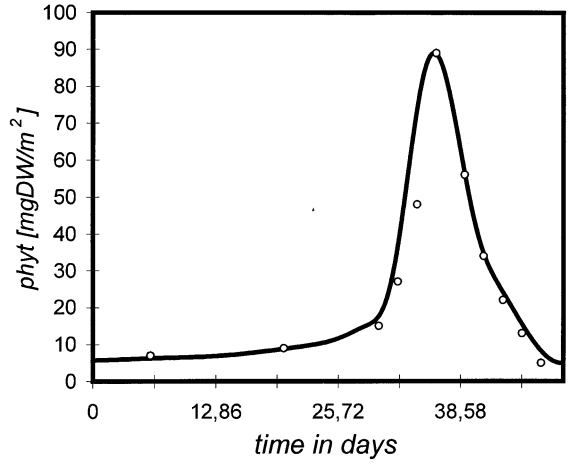

Fig. 3. New data set for phytoplankton growth.

new data set. The linear equation found by LAGRAMGE was:

$$\dot{\text{phyt}} = -5.41 - 0.0439 \cdot \text{phyt} - 13.5 \cdot \text{nitro} - 38.2 \cdot \text{zoo} + 93.9 \cdot \text{phosp} + 3.20 \cdot \text{temp}$$

The graph in Fig. 4 shows the correlation coefficients between the measured values and the values predicted with four different predictors for new data set as the prediction period increased. We can observe a significant improvement of both

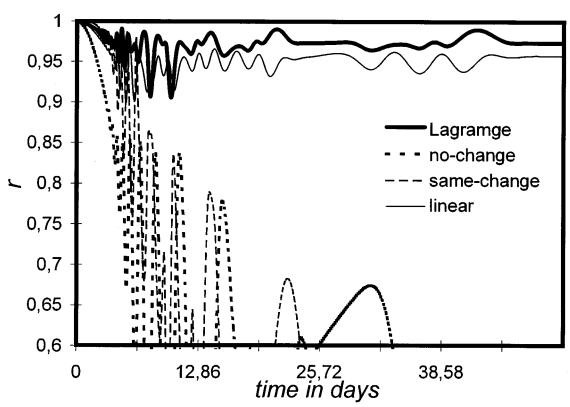

Fig. 4. Dependence of correlation coefficients between the measured values and values predicted with the four different predictors on increasing prediction period for the new experimental data set.

accuracy and robustness of the predictor obtained with LAGRAMGE. The linear predictor has accuracy comparable to, but lower than the accuracy of the LAGRAMGE predictor. Also the linear modell is less comprehensible for the human experts as it does not use background knowledge. The predictor obtained with LAGRAMGE is now also suitable for long-term prediction of phytoplankton growth in Lake Glumsoe.

5. Discussion

We presented the equation discovery system LAGRAMGE that uses declarative language bias for restricting the hypothesis space of equations. Within the declarative language bias formalism presented in the paper, background knowledge in the form of function definitions can be used along with common arithmetical operators and functions built in the C programming language. The hypothesis space of LAGRAMGE is a set of equations, such that the expressions on their right hand sides can be derived from a given context free grammar.

In contrast with other system identification methods, where the structure of the model has to be provided explicitly by the human expert, LAGRAMGE can use a more sophisticated form of representing the expert's theoretical knowledge about the domain at hand. A context free grammar can be used to specify a whole range of possible equation structures that make sense from the expert's point of view. Therefore, the discovered equations are in comprehensible form and can give domain experts better or even new insight into the measured data. This distinguishes LAGRAMGE from other methods for automated modelling like neural networks and polynomial regression, which can be used for obtaining black-box models, i.e. models with an incomprehensible structure.

The main advantage of declarative bias as compared to built-in language bias used in other equation discovery systems is that the user is allowed to adapt the learning system to the domain at hand. The language biases used in existing equation discovery systems restrict their hypothesis spaces to some manageable small hypothesis space of equations, which often affects the performance of the system (the correct equation may easily be out of the hypothesis space) as well as the comprehensibility of discovered equations (equations equivalent to the expected ones are discovered, but they are written in different form). If the constraint of comprehensibility is applied, one may lose some accuracy on the training data, but if accuracy is paramount the expert can use a different, less restrictive grammar (e.g. specifying polynomial equations as in LAGRANGE and GOLDHORN). Another major point of using background knowledge is the possibility to focus the search in the space of possible equations, which allows discovery of equations with complex structure that were out of reach for previous equation discovery systems (Todorovski and Džeroski, 1997). Finally, restricting the hypothesis space prevents the overfitting of the equation to the training data and the degradation of the model's accuracy on test cases not available to the learning system during the learning process.

If plenty of measurement data are available, less restrictive bias (more general equation space) can be used. On the other hand, when less data are available, more background knowledge should be used to restrict the space of equations and compensate for the lack of data. While classical identification methods consider one equation structure and other equation discovery methods consider a large space of equations regardless of the amount of data available, LAGRAMGE offers the possibility to trade-off between measurement data and background knowledge. The size of the data set is one aspect of the data quality. The trade-off mentioned above applies to the overall quality of data. For example, where noise would cause a nonsense model to be selected based on fit, background knowledge can restrict the space of models to avoid such models. LAGRAMGE in itself has no special noise handling procedures for discovery from noisy data. Note, however, that background knowledge reduces the effect of noise, as discussed above.

If several models have approximately the same goodness of fit, LAGRAMGE would by default choose the shortest one, due to the MDL heuristic function. Note that at the end of its search

through equation space, LAGRAMGE actually reports not only the best equation (according to the heuristic function), but a number of equations (the actual number is selected by the user by setting a parameter of LAGRAMGE). This would allow a domain expert to opt for an equation with slightly worse fit which agrees better with expert opinion.

Equations discovered by LAGRAMGE can be expected to be valid for the system variables value ranges encountered in the training data. We do not explicitly present these ranges together with the equations, but they can be extracted easily from the training data. It may happen that a model that describes the behavior of a dynamic system well for one range is not the best model if a different range is also considered. LAGRAMGE can take into account more than one measured behavior of a modelled system. In this way, it can be used for learning universal models of the observed system, which are valid on a wider range of values of the system variables. The probability of getting a good fit of the training data by chance is also reduced when considering more than one behavior.

The use of background knowledge was essential in the task of modelling phytoplankton growth in Lake Glumsoe on the basis of only 14 measurements within the period of two months. Even with such sparse measurement time points, LAGRAMGE discovered a comprehensible model which can be used successfully for predicting phytoplankton growth in the lake.

Expert knowledge was used in this domain at two different levels. Firstly, experts sketched the dynamic behavior of the observed system variables between the measurement points, which is regarded as additional source of reliable data. Secondly, a context free grammar was built using biological knowledge of population dynamics. The structure of the discovered equations tells us that phosphorus is a limiting factor for phytoplankton growth in the lake, along with temperature.

Even when loosing the comprehensibility constraint on discovered model (using context free grammar for linear equations), we obtained a standard linear model, discovered by LAGRAMGE using a context free grammar for linear equations, which is a less accurate predictor then the one obtained using the expert's context free grammar. It is also less comprehensible. We expect that the accuracy and robustness of the predictor, obtained with LAGRAMGE, can be improved by providing better measurement data to the process of equation discovery.

A direction for further work is to extend the declarative bias formalism to allow explicit definition of the constraints for the constant parameter values according to domain knowledge. More sophisticated parameter fitting procedures would have to be used to fit constant parameters of the equations according to these constraints. On the side of the experimental evaluation of LAGRAMGE domains should be addressed where extensive regular measurements over a long period of time are available. A comparison of LAGRAMGE with mainstream statistical methods for time series prediction (such as ARIMA) should be also done on such measurements.

Acknowledgements

This work was supported in part by the Slovenian Ministry of Science and Technology under the project ILP for Knowledge Discovery in Ecological Databases and by the ESPRIT IV project 20237 ILP2. We greatly appreciate the comments of two anonymous reviewers who substantially improved the paper. We would also like to thank Professor Sven-Erik Jørgensen for providing the Lake Glumsoe data.

References

Bendoricchio, G., Coffaro, G., DeMarchi, C., 1994. A traffic model for Ulva Rigida in the Lagoon of Venice. Ecol. Modelling 75/76, 485–496.

Crispi, G., Mosetti, R., 1993. Adjoint estimation of aquatic ecosystem parameters. Coenoses 8 (1), 11–14.

Dehaspe, L., De Raedt, L., 1995. A declarative language bias for concept-learning algorithms. Knowl. Eng. Rev. 7 (3), 251–269.

Demšar, D., 1996. Experiments in automated modeling of ecological processes in Lake Glumsoe. BSc Thesis, Faculty of Computer and Information Science, University of Ljubljana, Slovenia (in Slovenian).

Džeroski, S., Todorovski, L., 1993. Discovering dynamics. Proceedings of the Tenth International Conference on Machine learning, Morgan Kaufmann, San Mateo, CA, pp. 97–103.

Hopcroft J.E., Ullman, J.D., 1979. Introduction to automata theory, languages, and computation. Addison-Wesley, Reading, MA.

Jørgensen, S.E., 1986. Fundamentals of Ecological Modelling, North-Holland, Amsterdam, pp. 118–119.

Jørgensen, S.E., Kamp-Nielsen, L., Chirstensen, T., Windolf-Nielsen, J., Westergaard, B., 1986. Validation of a prognosis based upon a eutrophication model. Ecol. Modell. 32, 165–182.

Kompare, B., 1995. The use of artificial intelligence in ecological modeling. PhD Thesis, Royal Danish School of Pharmacy, Copenhagen, Denmark.

Križman, V., Džeroski, S., Kompare, B., 1995. Discovering dynamics from measured data. Electrotech. Rev. 62, 191–198.

Ljung, L., 1993. Modelling of industrial systems. Proceedings of the Seventh International Symposium on Methodologies for Intelligent Systems, Springer, Berlin, pp. 338–349.

Monod, J., 1942. Recherches sur la croissance des cultures bacteriennes. Hermann, Paris (in French).

Nédellec, C., Rouveirol, C., Adé, H., Bergadano, F., Tausend, B., 1996. Declarative bias in ILP. In: De Raedt, L., (Ed.), Advances in Inductive Logic Programming, IOS Press, Amsterdam, pp. 82–103.

Press, W.H., Flannery, B.P., Teukolsky, S.A., Vetterling, W.T., 1986. Numerical Recipes, Cambridge University Press, Cambridge.

Todorovski, L., Džeroski, S., 1997. Declarative bias in equation discovery. Proceedings of the Fourteenth International Conference on Machine Learning, Morgan Kaufmann, San Mateo, CA, pp. 376–384.