Knowledge representation and qualitative simulation of salmon redd functioning. Part I: Qualitative modeling and simulation
Francois Guerrin, Jacques Dumas

To cite this version:

HAL Id: hal-02682696
https://hal.inrae.fr/hal-02682696
Submitted on 1 Jun 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Knowledge representation and qualitative simulation of salmon redd functioning. Part I: qualitative modeling and simulation

François Guerrin a,*, Jacques Dumas b

a Inra, Unité de Biométrie et Intelligence artificielle, BP 27, 31326 Castanet-Tolosan Cedex, France
b Inra, Station d’Hydrobiologie, BP 3, 64310 St-Pée-sur-Nivelle, France

Received 21 December 1999; received in revised form 3 October 2000; accepted 23 October 2000

Abstract

This work aims at representing empirical knowledge of freshwater ecologists on the functioning of salmon redds (spawning areas of salmon) and its impact on mortality of early stages. For this, we use Qsim, a qualitative simulator. In this first part, we provide unfamiliar readers with the underlying qualitative differential equation (QDE) ontology of Qsim: representing quantities, qualitative variables, qualitative constraints, QDE structure. Based on a very simple example taken of the salmon redd application, we show how informal biological knowledge may be represented and simulated using an approach that was first intended to analyze qualitatively ordinary differential equations systems. A companion paper (Part II) gives the full description and simulation of the salmon redd qualitative model. This work was part of a project aimed at assessing the impact of the environment on salmon populations dynamics by the use of models of processes acting at different levels: catchment, river, and redds. Only the latter level is dealt with in this paper. © 2001 Elsevier Science Ireland Ltd. All rights reserved.

Keywords: Qualitative simulation; Dynamical systems; Knowledge representation; Salmon redd ecology

1. Introduction

An important demand for models has emerged from both researchers and policy makers concerned with the environment. One of their major preoccupations is to predict and assess the impact of human activities on natural resources. Goals assigned to models are twofold.

1. Help better understand whole ecosystems properties through model analysis and simulation; and
2. help better manage them, namely by detecting malfunctions and identifying their causes, assessing the consequences of critical situations, reasoning and planning actions (all these tasks refer to system supervision).

However, modeling systems of interest for environmental sciences, be they an animal population, a forest, a lake, or an urbanized area, faces the
difficulty to involve a huge number of variables accounting for a great variety of processes of physical, chemical, biological, or social natures. The relations among those processes are generally incompletely and imprecisely known. To achieve the objectives of global understanding and management of these systems, it becomes necessary to place oneself at an observation level where an heterogenous amount of knowledge, scattered amongst various academic disciplines, must be integrated. For example, if one is interested in the management of migratory salmonid populations, once evidenced, on the one hand the influence of recruitment on the whole population dynamics, on the other hand filling and sealing of reds and stormflow events as main factors influencing the mortality of early stages, one must skip from the population level considered alone to consider the whole hydrological catchment to integrate all the factors affecting the functioning of reds. At that scale, obviously, one has to deal with a much higher degree of complexity and incompleteness. Relevant knowledge is scattered amongst domains as diverse as geography, climatology, geochemistry, soil sciences, terrestrial and river hydrology, forest ecology, agronomy, hydroecology, etc., and no more ichthyology and population dynamics only. This approach thus necessitates to go beyond the framework of “classical” models, which consider a small number of well delimited processes and account only with great difficulty for the whole set of interactions between subsystems relevant to our goals (Costanza et al., 1993).

The major issue of qualitative reasoning (QR) is to provide engineers or scientists (and among them, also, ecological scientists) with general formalizations allowing them to abstract the main relevant features of the complex real world. These methods (mostly based on symbolic ontologies) allow one to represent and integrate expert knowledge (most of time, imprecise and incomplete), and implement it as models with good self-explanatory facilities. At the opposite to rule-based expert systems, QR formalizations allow one to represent as concisely system features and behaviors as classical mathematical models do, but without needing much analytical description and numerical information about functions (a review of QR advantages for tasks such as decision-making can be found in Brajnik and Lines (1998)). Since a dozen of years, such models are being applied in various engineering domains to support tasks such as design, signal interpretation, fault detection and diagnosis (Weld and de Kleer, 1990; Kuipers, 1994; Travé-Massuyès et al., 1997). Beyond the great diversity of domains and tasks addressed, the QR community is also characterized by a great variety of methodological approaches such as:

- modeling ontologies, e.g., the component-connection approach (de Kleer and Brown, 1984) where a set of interconnected submodels represent the physical components of the real system, or the process-oriented approach (Forbus, 1984) based on the representation of causal influences acting on the objects of the modeled world;
- temporal abstraction, aiming at representing and simulating a hierarchy of processes exhibiting dynamics with different speeds (Kuipers, 1987; Ayrolles et al., 1996);
- compositional modeling of model fragments taken from libraries, that can be reused in various contexts (Falkenheiner and Forbus, 1991); and
- qualitative formal calculus: qualitative algebra, order of magnitude reasoning, interval algebra, etc. (a review of such formalisms may be found in Travé-Massuyès et al. (1997)).

Describing such a diversity is beyond the scope of this paper (some books provide a taste of that, like Travé-Massuyès et al. (1997)). Mainly based on informal knowledge (because of the lack of more precise information), we are interested in representing and simulating qualitatively the functioning of ecological systems. This work was motivated by a project aimed at assessing the impact of the environment on salmon populations by the use of models of processes acting at different levels: catchment (e.g., erosion), river (e.g., sediment transport and deposition), and reds2 (e.g.,

---

1 Classical expert systems necessitate an extensive description of the problem to solve as a set of I–Then rules. In complex cases, this generates huge knowledge bases with problems of tractability, maintenance, etc.

2 Areas where salmon reproduction occurs in the rivers.
fine particles infiltration and redd clogging). Here, we focus on the salmon redd system, with principal aim to predict and explain the survival rate of fish under various scenarios. Because taking into account the dynamics of processes was important and because of the continuous nature of most of relevant variables, we chose, amongst the various QR approaches, the Qualitative simulator Qsim (acronym of qualitative simulation; see Kuipers (1994)) as a modeling and simulation tool.

In this first part of the paper we provide unfamiliar readers with the Qsim ontology: quantity representation (Section 2), qualitative variables (Section 3), qualitative constraints (Section 4), qualitative differential equations (QDEs) structure (Section 5). Although Qsim was first intended to analyze qualitatively ordinary differential equations (ODE), we show how informal biological knowledge may also be represented and simulated with this tool. For this, we use a very simple example taken from the salmon redd application (sensitivity of fish to dissolved oxygen depletion). The full description and analysis of the salmon redd model is in the second part (Part II) of the paper.

2. Representing quantities

Qsim, like most of QR approaches, essentially deals with ordinal values to characterize quantities considered as attributes of physical objects with real (cardinal), but unknown, values. A quantity range, continuous by essence, is discretized by a set of totally ordered symbols (called landmarks), representing some relevant critical values the quantity can take. The set of landmarks defines a quantity space (qspace for short). Landmarks and open intervals between them partition the real-valued range of the quantity into qualitatively homogenous regions. For example, let us consider the series of biological events describing the development of salmon: egg (egg), eyed-embryo (eye), egg hatching (hat), alevin at the beginning (em₁) and the end (em₂) of emergence off the redd. As time goes on, but at unknown dates, salmon will pass successively through these stages that can be ordered chronologically: egg < eye < hat < em₁ < em₂. Since spawning occurs before all other event we can assign it a zero value (egg: = 0) and choose the positive domain [0, +∞) to account for all these events. Therefore, a relevant quantity space to describe the developmental stages of salmon can be the 6-symbols set: {0, eye, hat, em₁, em₂, +∞}. Open intervals (0, eye), (eye, hat), (hat, em₁), (em₁, em₂), (em₂, +∞), correspond to the following phases: egg to embryo development, alevin growth inside the redd, emerging of fry, further development of fry outside the redd, respectively. The ordered set made of the union of landmarks and open intervals between them is the range of qualitative magnitudes (qmag) the quantity can take. Note that the qspace definition is local and assume no relation with landmarks in other qspaces.

As Qsim is in Lisp, the qspace of stage (variable denoting the developmental stages of salmon) must be written in Lisp syntax:

\[
\text{(stage (0 eye hat em1 em2 inf))}, \tag{1}
\]

where inf denotes infinity. A much simpler quantity space could be (0 inf), except that it does not provide us with the relevant distinctions we want to set in salmon development.

Similarly, assuming the sensitivity of fish to dissolved oxygen concentration is a positive or null quantity that can be characterized by qualifiers such as low, medium, high (denoted by symbols lo, md, hi, respectively) a qspace for this variable (denoted Sens-02) can be

\[
\text{(Sens-02 (0 lo md hi inf))}. \tag{2}
\]

Note that, as any notion of distance between landmarks is lacking, their order only is known. On a real number scale, two consecutive landmarks may thus be very close or very far one

---

3 Qsim is developed at the University of Texas at Austin in Ben Kuipers’ QR Group. First published in 1986 (Kuipers, 1986), Qsim ontology and algorithm are extensively described in a more recent textbook in which interested readers are strongly advised to refer (Kuipers, 1994). Qsim implementation is available as a freeware for research purposes.

4 Default qspace is the widest range for variable values: (minf 0 inf), where minf denotes \(-\infty\).
from each other. However, their distance may be either unknown or makes no sense (how far is it to go from low to high, and back?). This incomplete information is precisely what justifies the use of QR formalizations.

3. Qualitative variables

A qualitative variable represents the time evolution of a quantity. It is expressed as a reasonable function of time, \( f(t) \), continuous on its definition domain \([t_0,t_n]\subset \mathbb{R}\) (a bounded temporal interval), continuously differentiable on \((t_0, t_n)\), with values in the extended reals (including infinite landmarks, i.e., \( \mathbb{R}\setminus \{-\infty, +\infty\}\)). The qualitative value of a variable, denoted \( QV(f,t) \), with qspace a set of landmarks \( l_1,\ldots,l_k \), is the pair \( \langle qmag(f),qdir(f) \rangle \) where \( qmag \) denotes the qualitative magnitude of the variable, and \( qdir \), its qualitative direction of change:\(^5\)

\[
qmag(f) = \begin{cases} l_i & \text{if } f(t) = l_i, \\ (l_i,l_{i+1}) & \text{if } f(t) = (l_i,l_{i+1}) \end{cases}
\]

\[
qdir(f) = \begin{cases} inc & \text{if } f'(t) > 0, \\ std & \text{if } f'(t) = 0, \\ dec & \text{if } f'(t) < 0. \end{cases}
\]

So, a \( qmag \) is either a landmark or a closed interval between two adjacent landmarks; a \( qdir \) denotes the sign of the variable’s time derivative. Like any quantity, the time of simulation is represented by ordered landmarks (instants) and intervals between them: \( t_0,t_1,t_2,\ldots,t_{n-1},t_n \). Its \( qdir \) is always increasing (i.e., \( inc \)). Therefore, the qualitative behavior of any variable is a sequence of qualitative values this variable takes over time. Returning to the example of the developmental stages of fish (expression 1), let us assume that: \( stage \) is monotonically increasing, that it is not influenced by any other variable, and that its simulation starts at spawning (i.e., \( QV(stage,t_0) = 0 \)). Then, the \( stage \) behavior will be described by the sequence

\[
\begin{align*}
QV(stage,t_0) &= \langle 0,inc \rangle \\
QV(stage,(t_0,t_1)) &= \langle 0,eye,inc \rangle \\
QV(stage,t_1) &= \langle eye,inc \rangle \\
QV(stage,(t_1,t_2)) &= \langle (eye,hat),inc \rangle
\end{align*}
\]

A distinguished time point in the behavior of a variable is an instant \( t_p \) such that \( f(t_p) = l_i \) (where \( l_i \) is a landmark in the variable’s qspace). In expression 3, \( t_0,t_1,\ldots \) are distinguished time points.

Note that, as the notion of distance between landmarks is lacking, so is the duration between time points. This can be a problem when dating is necessary. We will see in Part II how it is possible to skip, to some extent, from this problem.

Generalizing, a system is a set of variables. A system’s \( state \) at time \( t \), is the set of all the qualitative values of the variables it contains. A system’s \( behavior \) is described by the sequence of its qualitative states on \([t_0,t_n]\).

4. Qualitative constraints

4.1. General constraints

Relations among the variables of a system are named constraints. They are denoted in Qsim as predicates, using the Lisp syntax.

Main are the following:

\[
\begin{align*}
(\text{add } x y z) & \equiv \forall t \quad x(t) + y(t) = z(t) \\
(\text{mult } x y z) & \equiv \forall t \quad x(t) \times y(t) = z(t) \\
(\text{minus } x y z) & \equiv \forall t \quad x(t) - y(t) = z(t) \\
(\text{d/dt } x y) & \equiv \forall t \quad \frac{dx}{dt} = y(t), \\
(\text{constant } x) & \equiv \forall t \quad \frac{dx}{dt} = 0
\end{align*}
\]

In addition, there are functional relations, such as \( (M+ x y) \) and \( (M- x y) \), expressing that \( y(t) \) is an increasing \( (M+) \) or decreasing \( (M-) \) monotonic function of \( x(t) \). Notice that the precise form of these functions is unknown; \( M+ \) and \( M- \) constraints stand, in fact, for families of
monotonic functions (linear and non-linear) and express their common properties. For example, \((M + x y)\) implies that the derivatives \(x'(t)\) and \(y'(t)\) are the same sign (opposite signs in the case of \((M - x y)\)). It is also possible to specify non-monotonic constraints like \(U^+\) and \(U^-\) (U-shaped curves concave upward or downward, respectively), \(S^+\) and \(S^-\) (S-shaped curves, increasing or decreasing), monotonic multivariate constraints (generalizing \(M^+\) and \(M^-\)), or equational constraints (e.g., algebraic sums).

Assuming we have more knowledge at hand, these constraints may be made more precise by \(n\)-tuples of corresponding values. They are qualitative magnitudes (generally landmarks) known to satisfy the constraint. For example, the tuple \((0,0,0)\) gives corresponding values for the add constraint (actually it is implicit and need not be specified by the user). Specifying \((0,0)\) as corresponding values for \((M + x y)\) restricts the family of increasing monotonic functions to those passing through the origin of coordinates.

Although equivalences are set between qualitative constraints and operations or functions on the real line, their semantics is a bit different: it is based on the signs of the variables appearing in the constraint and their derivatives. Sign algebra is thus the basis for constraint checking; for details on sign algebra, see Kuipers (1994, chapter 3) or Trave-Massuyès et al. (1997, chapter 1)). For example, the add constraint implies that the time derivatives of its arguments are the same sign. Moreover, whether \((a,b,c)\) are corresponding values for this constraint, then for all \(t\)

\[
[x(t) - a] \oplus [y(t) - b] = [z(t) - c],
\]

where \([\cdot]\) denotes the sign \(s\) of the enclosed expression and \(\oplus\) is addition on signs.\(^6\) Similarly, \(((M - x y) (a b))\), denoting that \(y(t)\) is a decreasing monotonic function of \(x(t)\) and \((a, b)\) are corresponding values, implies that:

\[
[x'(t)] = -[y'(t)]
\]

\[
[x(t) - a] = -[y(t) - b]
\]

Of course, the classical algebraic rules on the product and its derivation apply:

\[
(mult x y z) \Rightarrow \\
\quad \sim [x(t)][y(t)] = [z(t)] \\
\quad \sim [x'(t)y(t)] \oplus [y'(t)x(t)] = [z'(t)].
\]

These implicit consequences are instantiated for all constraints under evaluation and are used as filters in constraint solving (Section 5.2).

### 4.2. Defining new constraints

In order to capture functions exhibiting ‘Hat’ shaped curves, we defined two new constraints \((H^+\) and \(H^-)\). Their semantics refer to the combination of two \(S\) constraints: e.g., \(H^-\) (concave downward) has the meaning of an \(S^+\) constraint followed, after a plateau (possibly reduced to an extremum point), by an \(S^-\) constraint. Let \(x\) and \(y\) be two qualitative variables, and \(x_1, x_2, x_3, x_4, y_1, y_2, y_3, y_4\), landmarks of the qspace of \(x\); \(y_1, y_2, y_3, y_4\), landmarks of the qspace of \(y\). Assuming \(x\) and \(y\) are linked by an \(H^-\) constraint, i.e., \(y = H^-(x)\), this constraint’s semantics is as follows (see also Fig. 1):

\[y = H^-(x)\]

\[
y = y_1 \quad y_2 = y_3 \quad y_4 = y_f
\]

\[
x = x_1 \quad x_2 \quad x_3 \quad x_4
\]

\[x_1, x_2, x_3, x_4, y_1, y_2, y_3, y_4\] landmarks of the qspace of \(x\); \(y_1, y_2, y_3, y_4\), landmarks of the qspace of \(y\). Assuming \(x\) and \(y\) are linked by an \(H^-\) constraint, i.e., \(y = H^-(x)\), this constraint’s semantics is as follows (see also Fig. 1):

\[^6\] \(s \oplus s = s \oplus 0 = s\) for \(s \in \{-0,+\}\), \(\oplus = \{-0,+\}\); \(\oplus\) is commutative.


\[ x \leq x_1 \Rightarrow \begin{cases} qdir(y) = \text{std}, \\ qmag(y) = y_1, \end{cases} \]

\[ x_1 < x < x_2 \Rightarrow \begin{cases} qdir(y) = \text{qdir}(x), \\ y_1 < qmag(y) < y_2 = y_3, \end{cases} \]

\[ x_2 \leq x \leq x_3 \Rightarrow \begin{cases} qdir(y) = \text{std}, \\ qmag(y) = y_2 = y_3, \end{cases} \]

\[ x_3 < x < x_4 \Rightarrow \begin{cases} qdir(y) = -qdir(x), \\ y_4 < qmag(y) < y_2 = y_3, \end{cases} \]

\[ x \geq x_2 \Rightarrow \begin{cases} qdir(y) = \text{std}, \\ qmag(y) = y_4, \end{cases} \]

where \( qmag(\cdot) \) and \( qdir(\cdot) \) are the qualitative magnitude and direction of change of the variable in brackets. \( H^+ \) constraint (concave upward) is derived symmetrically.

The Qsim syntax for this constraint is

\[(H - X Y (x1 y1)(x2 y2)(x3 y3)(x4 y4)),\]

where each \( (xi yi) \) pair is the landmark coordinates of bend points \( p_i \). Notice that one \textit{must} have \( y2 \equiv y3 \) (plateau); in addition one \textit{may} have \( x2 \equiv x3 \) (bend points \( p2 \equiv p3 \) if the plateau is restricted to a single point, i.e., a maximum). Also, the following order should hold among landmarks:

- \( X \)-axis: \( x1 < x2 < x3 < x4 \).
- \( Y \)-axis: \( y1 < y2 = y3 \) and \( y4 < y2 = y3 \) (whereas any order may be set between \( y1 \) and \( y4 \)).

For example, the relation between the sensitivity of fish to the lack of dissolved oxygen as a function of their developmental stage, i.e., \( Sens-O_2 \) vs. stage, can be computed as an \( H^- \) constraint

\[(H - \text{stage} Sens-02 (\text{eye} \text{lo})(\text{hat} \text{hi})(\text{hat} \text{hi})(\text{em1} \text{md})) \] (4)

Expression 5 states that fish sensitivity, low at spawning, begins increasing once the eyed embryo stage is reached, up to a maximum at egg hatching; then it decreases down to the beginning of emergence and stays at an intermediate value beyond (Fig. 2). This summarizes empirical knowledge of freshwater ecologists.

5. Qualitative differential equations

5.1. Structure of a qualitative differential equation

A QDE is a model, valid for some functioning mode of a real system. Formally, it can be expressed as a 4-tuple \( \langle V, Q, C, T \rangle \), where \( V \) is a set of variables, \( Q \), the set of associated qspaces, \( C \), the set of constraints holding among variables, and, possibly, \( T \), a set of transitions. Transitions are rules which conditional part checks whether the validity domain of the current QDE is reached and, possibly, fires the simulation of another QDE accounting for the new functioning mode.

Developing the example of sensitivity of fish to oxygen depletion taken above, a very simple QDE is presented in Fig. 3 using the Qsim syntax. It has only three variables (stage, its time derivative \( dstage \), and Sens-02), and three constraints. It is made of a Lisp macro, define-QDE (with arbitrary name QDE1), embodying two essential clauses: quantity-spaces, defining the names and the qspaces of the variables (as given by expressions 1 and 2), and constraints like \( H^- \) which relates stage to Sens-02 (see expression 5). Two optional clauses are added: text, allowing one to include a comment to be displayed on plots, and layout, organizing the display of plots on screen. Here, a simple transition is set, that halts the simulation whenever stage reaches the em2 landmark with an increasing qdir (see transition specification in the QDE1 code, Fig. 3). This is to say that the model validity vanishes when all the alevins have left the redd.

Fig. 2. Sensitivity of early stages of salmon to the lack of oxygen Sens-02 vs. stage, computed as an \( H^- \) constraint (0 on the stage axis denotes spawning).
5.2. Simulation of a qualitative differential equation

In addition, one must specify a simulation function (here called Sim-QDE1, see bottom of Fig. 3) that will, first, create an initial state (make-new-state) from its specification (assert-values), then simulate the QDE from this state using the Qsim algorithm (qsim) and, finally, display the plots (qsim-display).

Qualitative simulation aims at predicting all the possible qualitative behaviors of the system represented by a QDE. Main steps of the Qsim algorithm are (i) initial state completion, (ii) successor states derivation, and (iii) global filtering. Steps (i) and (ii) are performed by solving the QDE constraints. This means to assign qualitative values to free variables to complete an incompletely specified initial state, then to derive all the possible successor states from the current system’s state according to constraint satisfaction criteria (see constraints semantics based on signs, Section 4.1). Successor states derivation is made according to a set of succession rules based on continuity assumptions. E.g., any change in qualitative direction of a variable from inc to dec or dec to inc must pass through a state where the variable’s qdir is std (derivative becomes zero at an extremum). Two different categories of rules are used according to the cases:

7 Which is the case in our example where incomplete initial values of only two variables are provided.
**P-successor** rules: if the current time is an instant \( t_i \), then qualitative values are generated for the next time interval \( (t_i, t_{i+1}) \), and

**I-successor** rules: if it is a time interval \( (t_i, t_{i+1}) \), then qualitative values are generated for the next time point \( t_{i+1} \).

All the candidate successor states are then filtered out by solving the QDE constraints as described above (inconsistent states are eliminated). Global filtering is then used to filter out possible successor states that do not comply with some generic constraints (not specified in the QDE), e.g.: a new state must not be identical to its predecessor (no qualitative difference), simulation must halt whenever an equilibrium state is reached or a cyclic behavior is detected (the system passes through an already determined state) etc. New landmarks can be created automatically during a simulation to represent critical values in the qspaces of some variables.

Hence, this algorithm allows one to generate all the qualitative behaviors consistent with the QDE constraints. Therefore, from a single initial state, several behaviors can be derived (or none if no consistent initial state can be found out). Generating multiple behaviors is both an advantage and a problem. Advantage, because Qsim guarantees that any possible behavior will be necessarily predicted; the set of predicted behaviors covers all the qualitative descriptions of the solutions of an equivalent numerical model\(^8\) – proof of the guaranteed coverage theorem is in Kuipers (1994).

Problem comes from the fact that all the predicted behaviors are not necessarily possible. One speaks of spurious behaviors, in the sense that these are not solutions of an equivalent numerical model. A large piece of research has been devoted to reduce spurious behaviors generation by Qsim, but this is out of our purposes here; for discussion of that problem and possible solutions see Fouche (1992).

Getting back to our example, we show on Fig. 4 the tree representing the behavior generated by the simulation of QDE1. Its nodes correspond to the successive states that are predicted from the initial state (tree root). It exhibits a single behavior with 11 states corresponding to 6 time points (starting at initial state, ending at transition state) and 5 time intervals. The qualitative plots of the variable evolutions are displayed on Fig. 5. Although stage increases continuously from spawning at \( T_0 \) (like qualitative values listed in expression 3), since its time derivative is constantly positive, the sensitivity to oxygen remains steady at a low value from \( T_0 \) to \( T_1 \) (from spawning to the eyed-embryo stage), then starts growing between \( T_1 \) and \( T_2 \), passes by a maximum at \( T_3 \) at hatching (stage is hat), then decreases to stabilize at \( T_4 \) when the emergence of alevins begins (stage is em1). This behavior exhibits qualitatively the ‘Hat’ shaped curve specified by expression 4. It is consistent with Chapman’s (1988) conclusion according to which oxygen uptake by fish (and thus, its sensitivity to this factor) increases steadily from fertilization to hatching where it reaches a maximum. Examining the qualitative values of the variables at \( T_0 \) shows that all were instantiated with qmags and qdirs. Initial state completion was thus made, although two qmags only were specified as initial conditions (see Fig. 3, bottom).

Note that a new landmark has been created in dstage’s qspace: specifying as initial qmag a closed interval \((0, \infty)\) implies the existence of a positive landmark. Note also that, although a single state is needed to pass from one stage to another (from 0 to eye, hat to em1, and em1 to em2), three states are needed to transition from eye to hat. This is due to the Sens-02 variable, as, starting from lo at \( T_1 \), it needs crossing the intermediate md landmark to reach hi, which corresponds to change three times its state.\(^9\)

---

\(^8\) Note that any ODE may be ‘abstracted’ into a QDE as we will see in Part II; for principles see Kuipers (1994).

\(^9\) Crossing a landmark generates three states: before it, on it, and after it.
6. Discussion

We hope to have given the reader a taste of the ability of Qsim to allow for passing more or less directly from informal, incomplete, weak conceptual knowledge to an implemented model allowing for simulation. Moreover, qualitative simulation does not always result in a single solution, like ODEs, but gives all the possible trajectories of a system, thus improving our knowledge about it. In fact, a single QDE must be viewed as a family of many ODEs, differing not only by parameter real values but also by functional relations. This ability, though not shown here with the QDE1 example (where only one behavior was obtained), will be demonstrated in Part II.

However, some technical difficulties may arise from building a Qsim model such as determining consistent initial states, establishing relevant constraints among variables (namely in the cases where several alternatives seem possible), or specifying corresponding values in the constraints. Even the choice of qspace landmarks is not always that easy! All these choices have an impact, not on the general shape of the simulated behaviors but on their number. Sometimes the task can become a bit tedious. Let us emphasize again that Qsim was previously devised to qualitatively simulate abstractions of analytical model structures (ODEs) into qualitative models (QDEs). Previous existence of equations with analytical properties facilitates, indeed, the task of specifying a QDE, since the basic properties of the model may be derived from mathematical analysis; their translation into constraints is thus easier. In our case, it is a bit different: without any (or, at least, very few) equations, mainly based on informal expert knowledge, we try to formalize some model with analytical semantics. Therefore, it is advised to start from simpler forms of the model, with default qspaces and minimal set of constraints with very few corresponding values. Similarly, rather try to initialize simulations with minimally specified initial states. The result is generally to get too large a set of simulated behaviors with irrelevant distinctions. However, comparing these behaviors at branching states, through the automatic generation of variables provoking branching, enhances the knowledge about the QDE. This makes easier debugging and refining the QDE structure into more adequate versions: adding more constraints or modifying existing ones (like substituting an M+ monotonic constraint for a saturation function S+), introducing relevant corresponding values or necessary bend points for some constraints, and completing accordingly qspaces of the variables with necessary landmarks. Note that the goal is not necessarily to reduce the behavior set to a single one, but to eliminate uninteresting (or even spurious) behaviors. Interesting hints and techniques are given by Clancy et al. (1997).

Another way (that we have not tried yet) is that of compositional modeling, aimed at automatically generating a QDE from model fragments (generally consisting of the description of influences among a few variables, which is a natural way for

---

Fig. 5. Qualitative plotting of the single behavior given by QDE1 simulation (symbols used for plotting dots denote variable qdir values: and upward and downward arrows stand for inc and dec respectively, white dots stand for std).
expressing expertise in many domains) and a simulation scenario. Model generators like Qualitative process compiler by Farquhar (1994), thus generate corresponding QDEs and simulate them based on the Qsim algorithm. Such an approach has a strong interest in domains, like ours, where a large piece of knowledge must be integrated within the same model. It was applied in several domains, among which plant physiology (Rickel and Porter, 1992), terrestrial ecology (Salles and Bredeweg, 1997), as well as environmental economics (Brajnik and Lines, 1998). Main interest of this approach is that one can build large knowledge bases made of (reusable) model fragments, that are composed adequately with respect to the simulation scenario that is built (e.g., a what-if? question).

References

Weld, D., de Kleer, J. (Eds.), 1990. Readings in qualitative reasoning about physical systems, Morgan Kaufmann, San Mateo, CA, USA.