A new methodology for reasoning about semiqualitative dynamic systems

JUAN ANTONIO ORTEGA, RAFAEL M. GASCA, MIGUEL TORO
Departamento de Lenguajes y Sistemas Informáticos
University of Seville
Avda. Reina Mercedes s/n – 41012 – Sevilla
SPAIN

Abstract: - In science and engineering, knowledge about dynamic systems may be represented in several ways. The models are normally composed of quantitative, qualitative, and semiqualitative knowledge. All this knowledge should be taken into account when we study these models.

Different approximations have been developed when qualitative knowledge is taken into account: transformation of non-linear to piecewise linear relationships, MonteCarlo method, distributions of probability, causal relations, fuzzy sets, constraint logic programming, and combination of all levels of qualitative and quantitative abstraction.

In this paper, a new methodology for reasoning about dynamic systems with qualitative knowledge is proposed, This qualitative knowledge may be composed of: qualitative operators, envelope functions, qualitative labels and qualitative continuous functions. The formalism to incorporate this qualitative information into these systems is shown.

The main idea of the methodology follows: *when a semiqualitative model is transformed into a set of quantitative models, every quantitative model has a different quantitative behaviour, however, together, they may have similar qualitative behaviours*.

A brief description of the proposed methodology follows: a semiqualitative model is transformed into a set of quantitative models. The simulation of every quantitative model generates a trajectory in the phase space. A database is obtained with these quantitative behaviours. It is proposed a language to carry out queries about the qualitative properties of this database of trajectories. This language is also intended to classify the different qualitative behaviours of our model. This classification helps us to describe the semiqualitative behaviour of a system by means of hierarchical rules obtained by means of machine learning.

A theoretical study about the reliability of the obtained conclusions with the methodology is carried out.

In this paper, the methodology is applied to a logistic growth model with a delay. The evolution of bacteria, mineral extraction, world population growth, epidemics, rumours, economic developments, or learning curves are real-world systems whose behaviour patterns are closely related to a logistic growth.

The methodology will be applied in a real computer-controlled process. It is a production industrial system. A metallurgical Company is interested in modifying its steel control production system applying this methodology. They wish to improve the steel quality, and, if possible, reduce the production costs. This collaboration is now in a preliminary phase. In forthcoming papers, we will describe this system in detail and the conclusions we shall obtain.

Key-Words: - Semiqualitative simulation, Knowledge representation, Qualitative reasoning Proc.pp..4661-4666

1 Introduction

In science and engineering, models of dynamic systems are normally composed of quantitative, qualitative, and semiqualitative knowledge. All this knowledge should be taken into account when we study these models. Different levels of numeric abstraction have been considered in the literature. They may be a description: purely qualitative [4], semiqualitative[3], [5], numeric interval [7] and quantitative.

Different approximations have been developed in the literature when qualitative knowledge is taken into account: transformation of non-linear to piecewise linear relationships, MonteCarlo method, constraint logic programming, probability distributions, causal relations, fuzzy sets, and combination of all levels of qualitative and quantitative abstraction [2], [3].

We study dynamic systems with qualitative knowledge. The proposed methodology transforms a semiqualitative model into a family of quantitative models. A semiqualitative model may be composed of qualitative knowledge, arithmetic and relational operators, predefined functions *(ln,exp,sen,...)*, numbers and intervals.

2 The methodology

There is enough bibliography that studies stationary states, however, the study of transient states is also necessary. For example, it is very important in production industrial systems in order to optimise their efficiency. Both states of a semiqualitative dynamic system may be studied with the proposed methodology. The methodology is shown in Fig. 1.

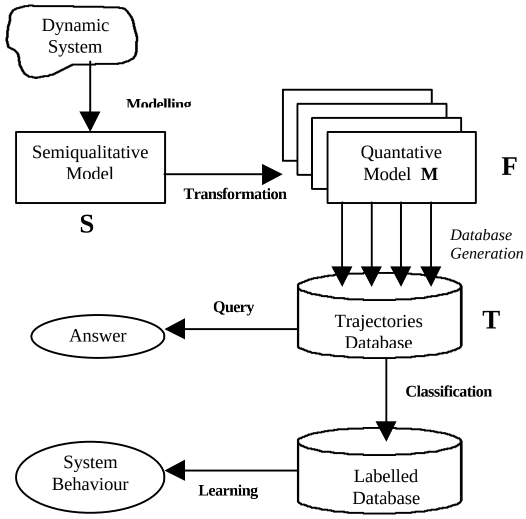

Fig. 1 Proposed methodology

Starting from a dynamic system with qualitative knowledge, a semiqualitative model *S* is obtained. A semiqualitative model *S* is represented by

$$\Phi(dx/dt,x,q,t),\ \ x(t_0) = x_0,\ \ \Phi_0(q,x_0) \qquad (1)$$

being $x \in \Re^n$ the set of state variables of the system, *q* the parameters, *t* the time, *dx/dt* the variation of the state variables with the time, Φ_0 constraints with the initial conditions, and Φ constraints on *dx/dt,x,q,t*.

If the methodology is applied, the equations of the dynamic system (1) will be transformed into a set of constraints among variables, parameters and intervals. In this paper, we are interested in those systems that may be expressed as (2) after transformation rules have been applied

$$dx/dt = f(x,p,t),\ \ x(t_0) = x_0,\ \ p \in I_p,\ \ x_0 \in I_0 \qquad (2)$$

where *p* includes the parameters of the system and new parameters obtained by means of the transformation rules, *f* is a function obtained by applying the transformation rules, and I_p,I_0 are real intervals. The equation (2) is a family *F* of dynamic systems depending on *p* and x_0.

A family of quantitative models *F* is obtained from *S* by applying some transformation techniques.

Stochastic techniques are applied to choose a model $M \in F$. This model *M* is quantitatively simulated obtaining a trajectory. It contains the set of values of all variables from their initial value until their final value, and the values of the parameters. Therefore, it contains the values of these variables in the transient and stationary states of the system. A database of quantitative trajectories *T* with these quantitative behaviours is obtained. A language is proposed in order to carry out queries about the qualitative properties of the set of trajectories included in the database. These trajectories may also be classified according to some criteria, and then, a labelled database is obtained. Qualitative behaviour patterns of the system may be automatically obtained from this database by applying machine learning based on genetic algorithms.

In the following sections, we are going to describe the steps of the methodology in detail.

3 Qualitative knowledge

Our attention is focused in those dynamic systems where there may be qualitative knowledge in their parameters, initial conditions and/or vector field. They constitute the semiqualitative differential equations of the system.

The representation of the qualitative knowledge is carried out by means of operators which have associated real intervals. The main advantage of this representation is that the integration of qualitative and quantitative knowledge is made in a simple way, and also facilitates the incorporation of expert knowledge in the definition of the range of the variables and the qualitative parameters of the system [2].

Qualitative knowledge may be composed of qualitative operators, qualitative labels, envelope functions and qualitative continuous functions. They and their transformation techniques are now detailed.

3.1 Qualitative operators

The representation of qualitative parameters and initial conditions is carried out by means of *qualitative operators*. Every qualitative operator *op* is defined by means of an interval I_{op} which is supplied by the experts. They may be unary *U* and binary *B* operators.

Each qualitative magnitude of the system has its own unary operators. Let U_x be the unary operators for a qualitative variable *x*, i.e. $U_x=\{VN_x, MN_x, LN_x, AP0_x, LP_x, MP_x, VP_x\}$. They denote for *x* the qualitative labels *very negative, moderately negative, slightly negative, approximately zero, slightly positive, moderately positive, very positive* respectively. The transformation rule for a unary operator is

$$op_u(e) \equiv \{r \in I_u\ ,\ e - r = 0 \qquad (3)$$

being r a new generated variable and I_u the interval associated with operator op_u. It is stablished in accordance with [6].

Let e_1,e_2 be two arithmetic expressions, and let op_b be a binary operator. The expression $op_b(e_1,e_2)$ denotes a qualitative order relationship between e_1 and e_2. *Binary qualitative operators* are classified into

- Operators related with the difference $\geq, =, \leq$. The following transformation rules are applied

$$\begin{aligned} e_1 = e_2 &\equiv \{ e_1 - e_2 = 0 \\ e_1 \leq e_2 &\equiv \{ e_1 - e_2 - r = 0,\ r \in [-\infty, 0] \qquad (4) \\ e_1 \geq e_2 &\equiv \{ e_1 - e_2 - r = 0,\ r \in [0, +\infty] \end{aligned}$$

- Operators related with the quotient *{«,−<,~,≈,»,Vo, Ne,...}*. The following transformation rule is applied

$$op_b(e_1,e_2) \equiv \{e_1 - e_2 * r = 0\ ,\ r \in I_b \qquad (5)$$

where I_b is the interval associated to operator op_b in accordance with [6].

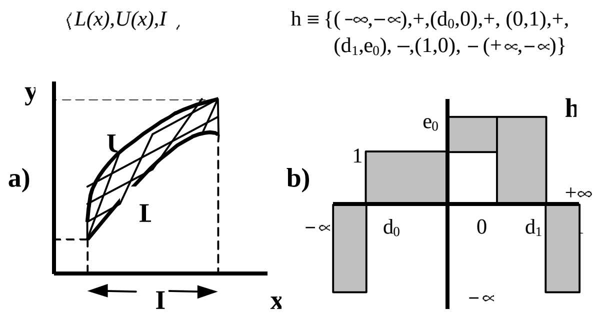

Fig. 2 Qualitative functions

3.2 Envelope Functions

An envelope function $y=g(x)$ (Fig. 2.a) represents the family of functions included between two defined real functions, a upper one $U:\ \Re \Rightarrow \Re$ and a lower one $L:\ \Re \Rightarrow \Re$. It is denoted by means of

$$\langle L(x),U(x),I\,\rangle, \qquad \forall x \in I: L(x) \leq U(x) \qquad (6)$$

where I is the definition domain of g, and x is the independent variable. The transformation rule applied to (6) is

$$g(x) = \alpha L(x) + (1 - \alpha)\, U(x) \text{ with } \alpha \in [0,1] \qquad (7)$$

where α is a new variable. If $\alpha=0 \Rightarrow g(x)=U(x)$ and if $\alpha=1 \Rightarrow g(x)=L(x)$ and any other value of α in (0,1) stands for any included value between $L(x)$ and $U(x)$.

3.3 Qualitative continuous functions

A qualitative continuous function $y=h(x)$ (Fig. 2.b) represents a constraint involving the values of y and x according to the properties of h. It is denoted by

$$y=h(x),\ \ h \equiv \{P_1,s_1,P_2,..,s_{k-1},P_k\} \qquad (8)$$

being P_i the points of the function. Every P_i is defined by a couple (d_i,e_i), being d_i the associated qualitative landmark to the variable x and e_i to y. These points are separated by the sign s_i of the derivative in the interval between two consecutive points. A monotonous qualitative function is a particular case where the sign is always the same $s_1=... =s_{k-1}$.

The qualitative interpretation (Fig. 3.a) for every $P_i=(d_i,e_i)$ of $y=h(x)$ is

$$\begin{aligned} &\text{if } x=d_i \Rightarrow y=e_i \\ &\text{if } d_i < x < d_{i+1} \Rightarrow \end{aligned} \begin{cases} s_i=+ \Rightarrow e_i < y < e_{i+1} \\ s_i=- \Rightarrow e_i > y > e_{i+1} \\ s_i=0 \Rightarrow y=e_i \end{cases} \qquad (9)$$

The transformation rules of a qualitative continuous function are applied in three steps:

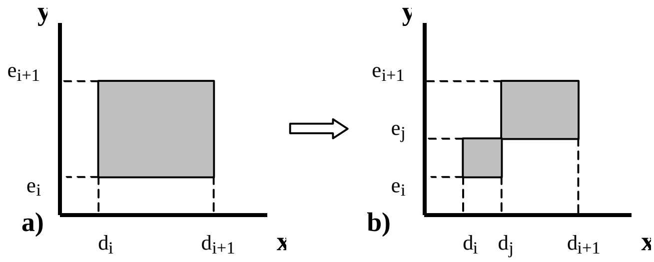

Fig. 3 Qualitative interpretation of a function

3.3.1 Normalization

The definition of the function is *completed* and *homogenised* using these continuity properties:

1. *a function that changes, its sign between two consecutive landmarks passes through a landmark whose value in the function is zero*
2. *a function whose derivative changes, its sign between two consecutive landmarks passes through a landmark whose derivative is zero*

The definition of any function (8) is always completed with: the extreme points ($-\infty$, $+\infty$), the points that denote the cut points with the axes, and where the sign of the derivative changes (a maximum or a minimum of h). New landmarks d_j,e_j keep an order relationship with the old ones (Fig. 3.b).

3.3.2 Extension

The definition of these functions is enriched by means of an automatic process. It incorporates new land-marks. We make it in order to dimish the uncertainty since the area of the rectangle is reduced (Fig. 3.b).

3.3.3 Transformation

A qualitative function

$$y=h(x),\ h \equiv \{P_1,s_1,P_2, ...s_{k-1},P_k\}\ \text{ with } P_i=(d_i,e_i) \quad (10)$$

is transformed into a set of quantitative functions H. The algorithm ***Choose H*** is applied to obtain H. It divides h into its segments. A *segment* is a sequence of consecutive points $\{P_m,...,P_n\}$ separated by means of those points whose landmark $e_i=0$ or where $s_i \neq s_{i+1}$.

The segments divide a function into monotonous regions where the landmarks e_i have the same sign. The algorithm applies stochastic techniques to choose every quantitative function of *H*. These techniques are similar to MonteCarlo method, however, the values obtained must satisfy the constraints of *h*. The followed heuristic applies a random uniform distribution to obtain the values for every landmark. The obtained values must verify the order relationship among their own landmarks.

4 Database generation

A family *F* of quantitative models has been obtained when the transformation rules have been applied to the semiqualitative model *S*. This family depends on a set of interval parameters *p* and functions *H* defined by means of a set of quantitative points. Every particular model *M* of *F* is selected by means of stochastic techniques, and it is quantitatively simulated. This simulation generates a trajectory *r* that is stored into the database *T*.

The following algorithms are applied to obtain *T*.

ChooseModel (F)

```
for every intervalar parameter or variable of F
      choose a value in its interval for it
for every function h of F
      choose a quantitative function H
```

Database generation T

```
T:={ }
for i=1 to N
    M:=ChooseModel(F)
    r:= QuantitativeSimulation(M)
    T:=T ∪ r
```

being *N* the number of simulations to be carried out. It is defined in accordance with the section *7*. Therefore, *N* is the number of trajectories of *T*.

5 Query/classification language

In this section, we propose a language to carry out queries on the trajectories database. The language is enriched with the ability to assign qualitative labels to the trajectories.

5.1 Abstract Syntax

Let *T* be the set of all trajectories *r* stored in the database. A query *Q* is: a quantifier operator $\forall, \exists, \aleph$ applied on *T*, or a basic query *[r,P]* that evaluates *true* when the trajectory *r* verifies the property *P*.

The property *P* may be formulated by means of the composition of other properties using the Boolean operators $\wedge, \vee, \neg$.

$Q: \forall\ r \in T \bullet [r,P]$	$P: P_b$	$P_b: P_d$
$\mid \exists\ r \in T \bullet [r,P]$	$\mid P \wedge P$	$\mid f(L(F))$
$\mid \aleph\ r \in T \bullet [r,P]$	$\mid P \vee P$	$\mid \forall t{:}F \bullet F$
$\mid [r,p]$	$\mid \neg P$	$\mid \exists t{:}F \bullet F$

A basic property P_b may be: a predefined property P_d, a Boolean function *f* applied to a list *L* of points or intervals that verifies the formula *F*, or a quantifier $\forall, \exists$ applied to the values of a particular trajectory for a time *t*. This time may be: an instant of time, a unary time operator (i.e. a range of time), a predefined time landmark, or the list of times where the formula *F* is verified.

A defined property P_d is the one whose formulation is automatic. They are queries commonly used in dynamic systems. There are two predefined: *EQ* is verified when the trajectory ends up in a stable equilibrium; and *CL* when it ends up in a cycle limit.

P_d: *EQ*	*F*: F_b	F_b: e_b
$\mid$ *CL*	$\mid F\ \&\ F$	$\mid e \in I$
	$\mid F \mid F$	$\mid u(e)$
	$\mid\ !\ F$	$\mid b(e,e)$

A formula *F* may be composed of other formulas combined by means of Boolean operators &,|,*!*.

A basic formula F_b may be: a Boolean expression e_b, or if a numeric expression *e* belongs to an interval, or a unary *u* or binary *b* qualitative operator.

Classification

A classification rule is formulated as a set of basic queries with labels, and possibly other expressions

$[r,P_A] \Rightarrow A, e_{n1}, ...$ $[r,P_b] \Rightarrow B, e_{n2}\}, ...$ ...

A trajectory *r* is classified with a label η if it verifies the property P_{r}.

5.2 Semantics

The semantics of every instruction of this language is translated into a query on the database. A query *[r,P]* is *true* when trajectory *r* verifies the property *P*. Semantics of a query with a quantifier depends on its related quantifier. If it is $\forall$, a Boolean value *true* is returned when all the trajectories $r \in T$ verify *P*. If it is $\exists$ then *true* is returned when there is at least one trajectory $r \in T$ that verifies the property *P*. If it is $\aleph$ then it is returned the number of trajectories of *T* that verifies *P*.

Let $\forall t{:}\ F_1 \bullet F_2$ be a basic property which is *true* if during the time that F_1 is satisfied, all the values of *r* verify F_2. For $\exists$ quantifier is *true* when at least a value of *r* that satisfies F_1, also satisfied F_2. In order to evaluate a formula *F*, it is necessary to substitute its variables for their values. These values are obtained from *T*.

Let $[r,P_A] \Rightarrow A,e_{A1}$ be a classification rule. A trajectory $r \in T$ is classified with the label *A* if it verifies property P_A. The result of evaluating e_{A1} for this trajectory is also stored into the database.

6 Theoretical study of the conclusions

When we claim "*there is a behaviour of the system that verifies the property P*" or "*all behaviours of the system verify the property P*", a question appears: are the obtained conclusions applicable to the real system? Its answer is proposed in this section.

The question to solve is: *what is the necessary condition to secure that all the behaviours of the system verify a property P?.*

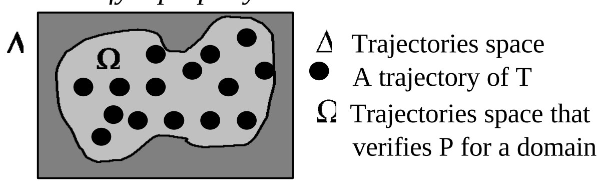

Fig.4 Trajectories space

Let Δ be the trajectories space of the system, an let Ω be the space of those trajectories of Δ that verify *P* (Fig. 4). Let *Vol(s)* be the volume of a space *s*. We are interested in knowing *what is the condition that should be verified in order to secure that* $Vol(\Delta)=Vol(\Omega)$*?*. In a schematic way, this question may be arise as what is the condition to secure that the next implication is true

$$\forall r \in T \bullet [r,P] \overset{\alpha}{\Rightarrow} \forall r \in \Delta \bullet [r,P] \qquad (11)$$

being α the confidence degree. Statistical techniques are necessary to carry out this implication.

Let *p* be the probability that a trajectory *r* verifies a property *Q* and $q=1-p$

$$p=Vol(\Omega)/Vol(\Delta) \qquad (12)$$

Let *x* be a random variable. For *n* trajectories the value *x* is *n* if the $n-1$ first trajectories verifies *Q*, and the $n-th$ does not.

Let α be the confidence degree. The expression

$$\alpha=P(x>n) \qquad (13)$$

is the probability that the *n* first trajectories verify *Q* and there is a trajectory that does not verify *Q* among the rest of trajectories of Δ.

Theorem: *The probability p verifies that*

$$p \geq 1-1/(n\alpha) \qquad (14)$$

Proof:

The expected value *E[x]* of a random variable *x* is defined as follows

$$E[x]=\sum_{n=1}^{\infty} n p^{n-1} q = q/p \sum_{n=1}^{\infty} n p^{n}$$

if it is replaced the geometric sum by its values

$$E[x]=q/p * p/(1-p)^2=1/(1-p) \qquad (15)$$

On the other hand, the Chevyshev inequality follows

$$E[x]=\sum_{x=1}^{\infty} xp(x) \geq \sum_{x=n+1}^{\infty} np^{n-1} = nP(x>n)$$

if *E[x]* is substituted by its value in (15), and if it is applied *(13)*, we obtain

$$E[x]/n \geq P(x>n) \equiv 1/(n(1-p)) \geq P(x>n)=\alpha$$

by means of symbolic manipulation the theorem is proved

$$1/n\alpha \geq 1-p \Rightarrow p \geq 1-1/(n\alpha) \qquad \square$$

This theorem proves that: *given an confidence degree* α*, if we want to ensure that a property P is true for a dynamic system with a probability p, it is necessary to obtain at least n trajectories verifying it.*

Next table shows several examples for *p* and *n* being $\alpha=0.05$ and $\alpha=0.01$

α=0.05		α=0.01	
p=0.6	n=50	p=0.5	n=200
p=0.8	n=100	p=0.9	n=1000
p=0.98	n=1000	p=0.99	n=10000
p=0.998	n=10000	p=0.999	n=10^6

In the same way, a query $\exists r \in \Delta \bullet [r,P]$ may always be formulated as $\neg \forall r \in \Delta \bullet [r, \neg P]$, applying a property of the predicate calculus, therefore, the previous study may also be applied for this quantifier.

7 A logistic growth model with a delay

It is very common to find growth processes where an initial phase of exponential growth is followed by another phase of asymptotic approach to a saturation value (Fig. 5.a). The following generic names are given: logistic, sigmoidal, and s-shaped processes. This growth appears in those systems where the exponential expansion is truncated by the limitation of the resources required for this growth. They abound in the evolution of bacteria, in mineral extraction, in world population growth, in epidemics, in rumours, in economic development, the learning curves, etc.

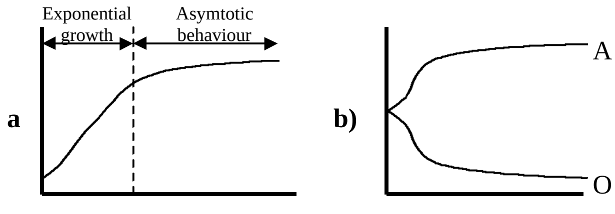

Fig. 5 Logistic growth model

In the bibliography, these models have been profusely studied. There is a bimodal behaviour pattern attractor: *A* stands for normal growth, and *O* for decay (Fig. 5.b).

Differential equations of the model *S* are

$$\Phi \equiv \begin{cases} dx/dt=x(n\,r-m),\ y=delay_{\tau}(x),\ x>0,\ r=h(y), \\ h \equiv \{(-\infty,-\infty),+,(d_0,0),+,\ (0,1),+, \\ \quad (d_1,e_0),\ -,(1,0),\ -\ (+\infty,-\infty)\} \end{cases} \quad (16)$$

being *n* the increasing factor, *m* the decreasing factor, and *h* a qualitative function with a maximum point at (x_1,y_0) (Fig. 2.b). The initial conditions are

$$\Phi_0 \equiv \{\ x_0 \in [LP_x,MP_x],\ LP_x(m),LP_x(n),\ \tau \in MP_{\tau},VP\ \}$$

where *LP*,*MP*,*VP* are qualitative unary operators for x,τ variables.

We would like to know:

1. if an equilibrium is always reached
2. if there is an equilibrium whose value is not zero
3. if all the trajectories with value zero at the equilibrium are reached it without oscillations.
4. To classify the database in accordance with the behaviours of the system.

We apply our methodology to this model. Firstly, the transformation rules to *S* are applied,

$$\Phi \equiv \begin{cases} dx/dt=x(n\,r-m),\ y=delay_{\tau}(x),\ x>0,\ r=H(y), \\ H,\ x_0 \in [0,3],\ \ m,n \in [0,1],\ \tau \in [0.5,10] \end{cases} \quad (17)$$

where *H* has been obtained by applying *Choose H* to *h*, and the intervals are defined in accordance with the experts knowledge. The algorithm *Database generation T* returns the trajectories database.

The proposed queries are formulated as follows:

1. $\forall r \in T \bullet [r,EQ]$
2. $\exists r \in T \bullet [r,\ EQ \wedge \exists t\!:\ t \approx t_f \bullet\ !AP0_x(x)]$
3. $\forall r \in T \bullet [r,\ EQ \wedge \exists t\!:\ t \approx t_f \bullet\ AP0_x(x) \wedge length(dx/dt=0)=0\ ]$

being $AP0_x$ a unary operator of *x*. The list of points where *dx/dt=0* is the list with the maximum and minimum points. If length is *0* then there is not oscillations.

We classify the database by means of the labels:

$$[r,EQ \wedge length(dx/dt=0)>0 \wedge \exists t : t \approx t_f \bullet\ !AP0_x(x)] \Rightarrow recovered,$$

$$[r,EQ \wedge length(dx/dt=0)>0 \wedge \exists t : t \approx t_f \bullet AP0_x(x)] \Rightarrow retarded,$$

$$[r,EQ \wedge \exists t : t \approx t_f \bullet AP0_x(x)] \Rightarrow extinction,$$

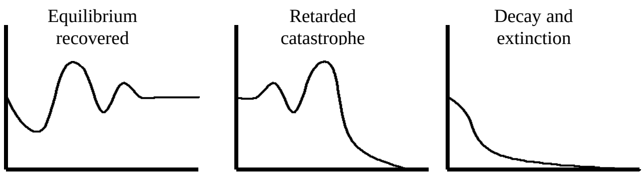

Fig. 6 Logistic growth model with a delay

They correspond to the three possible behaviour patterns of the system (Fig. 6). They are in accordance with the obtained behaviours when a mathematical reasoning is carried out [1].

8 Conclusion and further work

In this paper, a methodology is presented in order to automatize the analysis of dynamic systems with qualitative and quantitative knowledge. This methodology is based on a transformation process, application of stochastic techniques, quantitative simulation, generation of trajectories database and definition of a query/classification language.

The simulation is carried out by means of stochastic techniques. The results are stored in a quantitative database. It may be classified by means of the proposed language. Once the database is classified, genetic algorithms may be applied to obtain conclusions about the dynamic system.

In the future, we are going to enrich the query/classification language with: operators for comparing trajectories among them, temporal logic among several times of a trajectory, more type of equations, ...

A metallurgical Company is interested in modifying its steel control production system applying the proposed methodology. In forthcoming papers, we will describe this system in detail and the conclusions we shall obtain.

References:

[1] Aracil J., Ponce E., and Pizarro L., Behavior patterns of logistic models with a delay, *Mathematics and computer in simulation* No 44, 1997, pp. 123 - 141.

[2] Gasca R.M., Razonamiento y Simulación en Sistemas que integran conocimiento cualitativo y cuantitativo, *Ph.D. diss., Seville University*, 1998.

[3] Kay H., Refining imprecise models and their behaviors. *Ph.D. diss., Texas University*, 1996.

[4] Kuipers B.J. *Qualitative reasoning. Modeling and simulation with incomplete knowledge*, The MIT Press, 1994.

[5] Ortega J.A., Gasca R.M., and Toro M., Including qualitative knowledge in semiqualitative dynamic systems. *Lecture Notes in Artificial Intelligence* No.1415, 1998, pp. 329 - 338.

[6] Travé,-Massuyès L., Dague Ph., and Guerrin F., *Le raisonement qualitativ pour les sciences de l'ingénieur*, Hermes Ed., 1997.

[7] Vescovi M., Farquhar A., and Iwasaki Y., Numerical interval simulation: combined qualitati-ve and quantitative simulation to bound beha-viours of non-monotonic systems. *Proceedings of 14th International Joint Conference on Artificial Intelligence*, 1995, pp. 1806-1812.