# Improving Semi-Quantitative Reasoning by Landmark Approximation

**Karsten R. Heidtke**
Max-Planck-Institute for Molecular Genetics
Department Lehrach,
Ihnestraße 73
D-14195 Berlin, Germany
heidtke@mpimg-berlin-dahlem.mpg.de

**Steffen Schulze-Kremer**
Resource Center of the
German Human Genome Project
Heubnerweg 6
D-14059 Berlin, Germany
steffen@rzpd.de

**Abstract**

Many extensions to existing qualitative simulation packages allow combined use of available numeric and qualitative information. Some provide numeric integration across two qualitative states but none of them highlights numerically or semi-quantitatively the qualitatively predicted time points. Moreover, high accuracy at time points is only achieved by a fine level of granularity across the entire time interval. This paper presents landmark approximation as a new methodology to improve semi-quantitative predictions in qualitative simulation. We also present a transition table and a method to propagate semi-quantitative values that is suitable for landmark approximation. These techniques have been implemented in a simulation engine that is based on qualitative reasoning techniques with a semi-quantitative extension. We demonstrate that a semi-quantitative approximation to landmarks improves the predictive power of the simulation tool even with a reduced number of interpolated states.

## Introduction

Qualitative reasoning techniques, as introduced e.g. by (de Kleer & Brown 1984; Forbus 1984; Kuipers 1986), have been developed to model and simulate physical processes. It is a technique that is capable to describe relations in terms of qualitative equations or constraints for simulation. Such constraints cover a set of numerical solutions of a qualitative model but may also produce spurious behaviours that are not consistent with any numerical solution. Some extensions to qualitative simulation allow the combination of qualitative and quantitative information to prune spurious behaviours and provide quantitative predictions. Others make predictions over the qualitative time interval. These predictions are in some cases very weak because the distance between two time points can be very large. Another set of programs overcomes these weak predictions by interpolation of new time points. On the one hand this increases the precision of quantitative predictions but on the other these approaches may require a large number of refinement steps to reach a fine level of granularity, which can be a prohibitively time consuming task. However, since a fine level of accuracy can only be achieved across an entire interval, we propose landmark approximation to gain more accuracy at the possibly interesting time points. A fixed step size can be chosen for numeric integration and at the interesting time points we reach a fine level of granularity by approximating the time point to a chosen $\varepsilon$. Furthermore, we present a transition table that defines a set of possible state transitions, which is suitable to landmark approximation.

The presented work extends the simulation engine of the qualitative simulation environment for molecular biology *BioSim* (Heidtke & Schulze-Kremer 1998) which is based on qualitative descriptions with a semi-quantitative extension. The system provides modelling features such as objects and processes. The object structure allows to constrain parameters of active objects and the process structure allows to define constraints on object parameters that hold under certain preconditions. Given such a description the algorithm performs repeatedly a qualitative analysis of the current object parameters of a given state and predicts possible qualitative successor states. If quantitative information has been made available, new time points are interpolated, controlled by the qualitative prediction and refined by the landmark approximation algorithm at qualitatively predicted time points.

## System and Methods

The underlying simulation engine is a constraint centered approach providing an object and process structure. The characteristics of an object are described by parameters that are functions over time and are represented by a set of ordered landmarks and intervals between them, termed quantity space. This purely symbolic description can be extended by available semi-quantitative information. A semi-quantitative magnitude is either a numeric magnitude or an interval in the range of two numeric magnitudes. A landmark can be bound to a numeric value or a numeric interval. If the landmark is bound to an interval its borders are included. The value of a parameter includes at minimum a magnitude and a direction of change that can be extended by a semi-quantitative magnitude. Figure 1 shows an example of two objects.

**Example 1** *In Figure 1, the initial qualitative value* `(v,dec)` *of parameter* `velocity` *in object* `rocket` *could be represented as a semi-quantitative value* `([3000,3300],v,dec)`*, too. In this context both notations have the same meaning, because landmark* `v` *is bound to the interval* [3000, 3300] *in the quantity space.*

```
object( earth,
  'Earth',
  [ [ mass,
      [0,  m ~  5.98e24,  inf],
      [(m, std)] ],
    [ gravitation,
      [-inf, g ~ -6.67e-11, 0],
      [(g, std)] ],
    [ surface,
      [0,  sl ~ 6.37e6,   inf],
      [(sl,std)] ] ],
  [ constant( surface ),
    constant( gravitation ),
    constant( mass ) ]
).

object( rocket,
  'with initial velocity in decr. gravity',
  [ [ dist,
      [0,inf],
      [(0,inc)] ],
    [ accel,
      [-inf,'-g',0,inf],
      [('-g',inc)] ],
    [ radius,
      [0,inf],
      [((0,inf),inc)] ],
    [ velocity,
      [-inf,0,v ~ [3000,3300] ,inf],
      [(v,dec)] ] ],
  []
).
```

Figure 1: The semi-quantitative model describes two objects: `earth` and `rocket`. The `earth` is characterised by `mass`, `surface` (distance from the earth's core) and `gravitation`. Each of the parameter's quantity space has a landmark that is bound to a numeric value, e.g. `mass` has the landmark `m` bound to $5.98 \times 10^{24}$. All parameters are defined to be constant. The `rocket` is characterised by the distance `dist` from the earth's surface, the `radius` from the earth's core, `velocity` and acceleration `accel`. In the quantity space of `velocity` the landmark `v` is bound to the interval [3000, 3300]. Units have intentionaly been left out.

Processes describe functional dependencies between object parameters. Process preconditions determine whether or not a process holds, i.e. it is active or inactive. An example of a process defining constraints on objects is given in Figure 2. The qualitative behaviour of a model is derived from the directions of change of a parameter in conjunction with its qualitative and its semi-quantitative magnitude, if available. Possible transitions from one state to another are determined by a transition table. The set of possible qualitative behaviours is then represented by one or more behaviour trees. Each node represents a state of the entire system described by all values of object parameters at one point in time.

Functional dependenices can be defined in objects on their own parameters and in processes on parameters across objects. The system provides three different types of constraints:

```
process( acceleration,
  'Acceleration of rocket',
  [ rocket(dist) ge 0 ],
  [ add(earth(surface),rocket(dist),
       rocket(radius)),
    'm+'(rocket(dist), rocket(accel),
       [(inf,0),(0,'-g')]),
    rocket(accel) =
         (earth(gravitation)*earth(mass))/
         (rocket(radius)*rocket(radius)),
    'd/dt'(rocket(dist), rocket(velocity)),
    'd/dt'(rocket(velocity),rocket(accel)) ]
).
```

Figure 2: The semi-quantitative model describes one process `acceleration` that describes the functional dependencies in a qualitative (`add`, `'m+'` and `'d/dt'`) and quantitative (`'='`, `add`, `'d/dt'`) manner. This process is only active as long as parameter `dist` of object `rocket` is greater or equal 0.

- Qualitative constraints: `m+`, `m-`, `constant`, `increasing` and `decreasing`.
- Qualitative-semi-quantitative constraints: `add`, `sub`, `mult`, `div` and `d/dt`.
- Semi-quantitative constraint: `'='`. The right hand formula can be any combination of addition, subtraction, multiplication and division.

The implemented numeric constraint solver can handle real and interval arithmetic (Alefeld & Herzberger 1983; Struss 1990; Hyvönen 1992), with respect to multiplication, division, addition and subtraction. For numeric integration the simulation engine provides three different algorithms:

- Euler's method,
- Runge Kutta $2^{nd}$ order and
- Runge Kutta $4^{th}$ order with adaptive step size control.

The following two subsections describe in more detail the transition table of the simulation engine and the method of propagting semi-quantitative information on qualitatively predicted time points.

## Transition table

The behaviour of a system is described by changes of the qualitative value of parameters. The manner of changing the qualitative value of a parameter while changing from one state to another is restricted by a set of state transitions. The transition table that is given in the following definition only allows state transitions from one distinguished time point to another. State transitions from a time point to a time interval or vice versa are not allowed. Therefore, all changes are identifiable at distinguished time points.

**Definition 2** *The transition table of BioSim.* qv(t) *represents a qualitative value at a point in time* t. $l_j$ *represents a landmark,* $(l_j, l_{j+1})$ *an interval between two neighbouring landmarks in a quantity space.*

**transitions from steady values**

| | $qv(t_n)$ | $\Rightarrow$ | $qv(t_{n+1})$ |
|---|---|---|---|
| 1) | $(l_j, std)$ | | $(l_j, std)$ |
| 2) | $(l_j, std)$ | | $(l_{j+1}, inc)$ |
| 3) | $(l_j, std)$ | | $((l_j, l_{j+1}), inc)$ |
| 4) | $(l_j, std)$ | | $(l_{j-1}, dec)$ |
| 5) | $(l_j, std)$ | | $((l_j, l_{j-1}), dec)$ |
| 6) | $((l_j, l_{j+1}), std)$ | | $((l_j, l_{j+1}), std)$ |
| 7) | $((l_j, l_{j+1}), std)$ | | $(l_{j+1}, inc)$ |
| 8) | $((l_j, l_{j+1}), std)$ | | $((l_j, l_{j+1}), inc)$ |
| 9) | $((l_j, l_{j+1}), std)$ | | $(l_j, dec)$ |
| 10) | $((l_j, l_{j+1}), std)$ | | $((l_j, l_{j+1}), dec)$ |

**transitions from increasing values**

| | $qv(t_n)$ | $\Rightarrow$ | $qv(t_{n+1})$ |
|---|---|---|---|
| 11) | $(l_j, inc)$ | | $(l_{j+1}, inc)$ |
| 12) | $(l_j, inc)$ | | $(l_{j+1}, std)$ |
| 13) | $(l_j, inc)$ | | $((l_j, l_{j+1}), inc)$ |
| 14) | $(l_j, inc)$ | | $((l_j, l_{j+1}), std)$ |
| 15) | $((l_j, l_{j+1}), inc)$ | | $(l_{j+1}, inc)$ |
| 16) | $((l_j, l_{j+1}), inc)$ | | $(l_{j+1}, std)$ |
| 17) | $((l_j, l_{j+1}), inc)$ | | $((l_j, l_{j+1}), inc)$ |
| 18) | $((l_j, l_{j+1}), inc)$ | | $((l_j, l_{j+1}), std)$ |

**transitions from decreasing values**

| | $qv(t_n)$ | $\Rightarrow$ | $qv(t_{n+1})$ |
|---|---|---|---|
| 19) | $(l_j, dec)$ | | $(l_{j-1}, dec)$ |
| 20) | $(l_j, dec)$ | | $(l_{j-1}, std)$ |
| 21) | $(l_j, dec)$ | | $((l_{j-1}, l_j), dec)$ |
| 22) | $(l_j, dec)$ | | $((l_{j-1}, l_j), std)$ |
| 23) | $((l_j, l_{j+1}), dec)$ | | $(l_j, dec)$ |
| 24) | $((l_j, l_{j+1}), dec)$ | | $(l_j, std)$ |
| 25) | $((l_j, l_{j+1}), dec)$ | | $((l_j, l_{j+1}), dec)$ |
| 26) | $((l_j, l_{j+1}), dec)$ | | $((l_j, l_{j+1}), std)$ |

The following gives an example how to interpret the table:

**Example 3** *Assuming a quantity space* $[0, a, inf]$ *and a qualitative value* $(a, inc)$ *at time point* $t_2$ *the possible successor values at time point* $t_3$ *are determined from 11), 12), 13) and 14) to be* $(inf, inc)$, $(inf, std)$, $((a, inf), inc)$ *and* $((a, inf), std)$.

When a parameter changes to or from a steady state, the direction of change in the interval is determined by the direction of change at the time point that succeeds or preceeds the current time point, respectively.

## Propagation of quantitative information

In this section we describe how numeric, interval and qualitative information is combined and propagated. We answer an important question: "What is the quantitative scope of a symbolic magnitude?" in Definition 4. In the following, we use the notations:

- $sqm(p(t))$ to refer to the semi-quantitative magnitude,
- $qm(p(t))$ to refer to the symbolic magnitude and
- $qdir(p(t))$ to refer to the qualitative direction of change

of the parameter $p$ at a particular time $t$.

**Definition 4** *Let* $t_n$ *and* $t_{n+1}$ *be time points of two successive qualitative states, let nm be a newly computed semi-quantitative magnitude for a parameter p.*

1. *If* $p(t_{n+1})$ *is a qualitative value then nm extends the qualitative value to a semi-quantitative value, iff the* $qm(p(t_{n+1}))$ *is not bound to a semi-quantitative magnitude within the parameters quantity space.*
2. *If* $p(t_{n+1})$ *is a semi-quantitative value or a qualitative value, where the symbolic magnitude is bound to a semi-quantitative magnitude within the parameters quantity space, then*
   (a) *if* $nm = [u, v] \wedge u, v \in \mathbf{R} \wedge u < v$ *and* $qm(p(t_{n+1}))$ *is bound to an interval* $[x, y] \wedge x, y \in \mathbf{R} \wedge x < y$ *and* $[u, v] \cap [x, y] = \emptyset$ *then the whole behaviour will be refused,*
   (b) *if* $nm \in \mathbf{R}$ *and* $qm(p(t_{n+1}))$ *is bound to* $x \in \mathbf{R}$ *and* $sqm(p(t_n)) \in \mathbf{R}$ *then the whole behaviour will be refused if* $x \notin [nm, sqm(p(t_n))] \wedge nm \leq sqm(p(t_n))$ *and* $x \notin [sqm(p(t_n)), nm] \wedge nm \geq sqm(p(t_n))$,
   (c) *if* $nm = [u, v] \wedge u, v \in \mathbf{R} \wedge u < v$ *and* $qm(p(t_{n+1}))$ *is bound to* $x \in \mathbf{R}$ *and* $x \notin [u, v]$ *then the whole behaviour will be refused,*
   (d) *if* $nm \in \mathbf{R}$ *and* $qm(p(t_{n+1}))$ *is bound to an interval* $[x, y] \wedge x, y \in \mathbf{R} \wedge x < y$ *and* $nm \notin [x, y]$ *then the whole behaviour will be refused.*
3. *The* $sqm(p(t_n))$ *and* $sqm(p(t_{n+1}))$ *must be consistent with the direction of change, e.g. if* $qdir(p(t_n)) = inc$ *then* $sqm(p(t_{n+1})) > sqm(p(t_n))$ *must hold.*

This definition guarantees, that two values $p(t_n)$ and $p(t_{n+1})$ at successive time points are consistent. The following gives an example to illustrate the above given definition.

**Example 5** *In our representation a landmark is a symbolic description that separates two qualitatively distinguished regions of the quantity space. The basic landmark* 0 *separates the region of positive values from the negative ones. When we extend a qualitative value,* $(0, inc)$*, with numeric information, this must be consistent with the existing symbolic magnitude* 0*, which is implicitly bound to the numeric magnitude* 0 *in the parameter quantity space.*

- *A newly computed numeric magnitude must match with the qualitative value* $(0, inc)$*, e.g.* 0 *would be appropriate to extend* $(0, inc)$ *to the semi-quantitative value* $(0, 0, inc)$*, where* 0 *is a semi-quantitative as well as a qualitative magnitude, inc is the direction of change. Definition 4.2.b.*
- *Any interval* $[x, y]$*,* $x, y \in \mathbf{R} \wedge 0 \in [x, y]$ *can be used to extend* $(0, inc)$ *to* $([x, y], 0, inc)$*, where* $[x, y]$ *is a semi-quantitative magnitude,* 0 *a qualitative magnitude and inc the direction of change. Definition 4.2.c.*

## Algorithm

Landmark approximation is a new method to refine the numeric integration at qualitative time points. These time points must be examined in fine detail if one wants to get numerically precise results. When interpolating a qualitatively predicted time interval, the algorithm detects whether one of the numeric magnitudes $sqm(p(t_n))$ or $sqm(p(t_{n+1}))$ lies outside the scope of the parameter's symbolic magnitude. In this case, the interpolated state is refused and the step is repeated with a halved step size, until a user chosen $\varepsilon$ is reached or the two successive magnitudes are within the same qualitative region.

Another criterion for landmark approximation is the direction of change. The direction of change is defined to be constant across a qualitative time interval. When the algorithm detects a change of direction of one of the parameter values, landmark approximation is applied, too. The following describes our algorithm of landmark approximation.

**Algorithm 6** *Given a set of numeric equations EQ, s the actual state, n the qualitative successor state of s, ms the maximum step size, mi the maximum number of interpolation steps, $\varepsilon$ to approximate numerically a qualitative landmark, let $h = \frac{ms}{mi}$ be the step size, let $ns = s$ be the initial state and let $nm = normal$ be the mode of the algorithm then the algorithm proceeds as follows:*

*S1) If $nm = normal \wedge h < ms$ or $nm = narrow \wedge h > \varepsilon$*

- *then let be $ps = ns$ the previously interpolated state.*
- *Otherwise the algorithm stops. The numeric magnitudes of state ns are propagated to the qualitative values of state n. If this fails the state n is marked as inconsistent(n).*

*S2) Apply the chosen numeric integration method on the set of equations EQ to interpolate the next state ns for the given step size h.*
*If the calculation fails, i.e. EQ has no solution, the state s is marked as $failN(s)$. The algorithm stops.*

*S3) Compare ns with the previously interpolated state ps.*
*If any newly interpolated magnitude falls outside the current qualitative region or if a change of direction has been detected then*

- *let be $h = \frac{h}{2}$ and $nm = narrow$.*
  *If $h > \varepsilon$ then go to Step 2, otherwise go to Step 1.*
- *Otherwise let be $nm = normal$ and go to Step 1.*

**Example 7** *Let $\{-inf,\ 0,\ +inf\}$ be the quantity space of parameter p, let $p(t_n) = (5,\ (0, +inf),\ dec)$ a decreasing semi-quantitative value at $t_n$ and $p(t_{n+1}) = (-2,\ (0, +inf),\ dec)$ its predicted successor value at $t_{n+1}$. The landmark 0 in the parameter's quantity space separates the positive and the negative numbers. The numeric magnitude $sqm(p(t_{n+1}))$ does not belong to the positive numbers and is rejected. In consequence, the step size is halved and the interpolation step for $t_{n+1}$ is repeated, until the interpolated state returns for $sqm(p(t_{n+1}))$ a positive magnitude or the step size is less than or equal to $\varepsilon$. When the stepsize becomes smaller than or equal to $\varepsilon$ landmark approximation stops. In that case the resulting $sqm(p(t_{n+1}))$ must match a qualitatively predicted state at $t_{n+1}$, as defined in Definition 4, or the behaviour will be refused.*

**Note 8** *A positive side effect of the landmark approximation can be observed when interpolating non-linear functions. Assuming a chosen step size for interpolation to be too wide to reflect the "real" function, particularly Euler's method will give false results for the interpolated time points. The algorithm detects when an interpolated state falls outside the current qualitative region of a parameter, the step size will be halved and the newly interpolated states are closer to the "real" function. The algorithm then proceeds with the newly computed step size.*

## Example

In this section we present an example to highlight the predictive power of the landmark approximation algorithm. The example is taken from (Berleant & Kuipers 1997) where it has been modelled using Q3, to be discussed later. We have chosen this example to compare our approach to a well established simulation engine.

The behaviour of a rocket, fired upwards from the earth's surface, is described by the follwing equation:

$$\frac{d^2r}{dt^2} = \frac{-GM}{r^2}. \tag{1}$$

The model that represents Equation 1 is given in Figures 1 and 2. The model is initialised with $G = -6.67 \times 10^{-11} \frac{m^3}{kg \times s^2}$ the gravitational constant and $M = 5.98 \times 10^{24} kg$ the earth mass. The rocket's initial position is at sea level $6.37 \times 10^6 m$, i.e. the distance from earth is $0m$. The rocket is fired upwards with an initial velocity within the range $[3000, 3300] \frac{m}{s}$ and the engine turned off. The initial velocity is less than the escape velocity, so that after some time the rocket will return back to earth.

We performed two simulation runs using Euler's method. The maximum number of integration steps has been set to 10 and 28, respectively. The third simulation run has been performed using Runge Kutta $4^{th}$ order with adaptive step size control. The maximum number of integration steps has been set to 10 and the maximum width of the time interval has been set to $500s$, i.e. the integration method will proceed in steps of $50s$. The $\varepsilon$ for landmark approximation has been set to $1s$. If a landmark is reached the landmark is approximated by dividing the step size by 2, until the step size is less than or equal to the given $\varepsilon$.

The results obtained are shown in Table 1. For comparison, we added the results from (Berleant & Kuipers 1997) and those of an iterative summation method. The table focuses on the time intervals $t_1$ and $t_2$. At $t_1$ the rocket reaches its maximum distance $max_D$ from the earth's surface and at $t_2$ the rocket returns to earth. The table's last row holds the number of interpolated states.

The Runge Kutta $4^{th}$ order method delivers the most precise result in comparison to the control val-

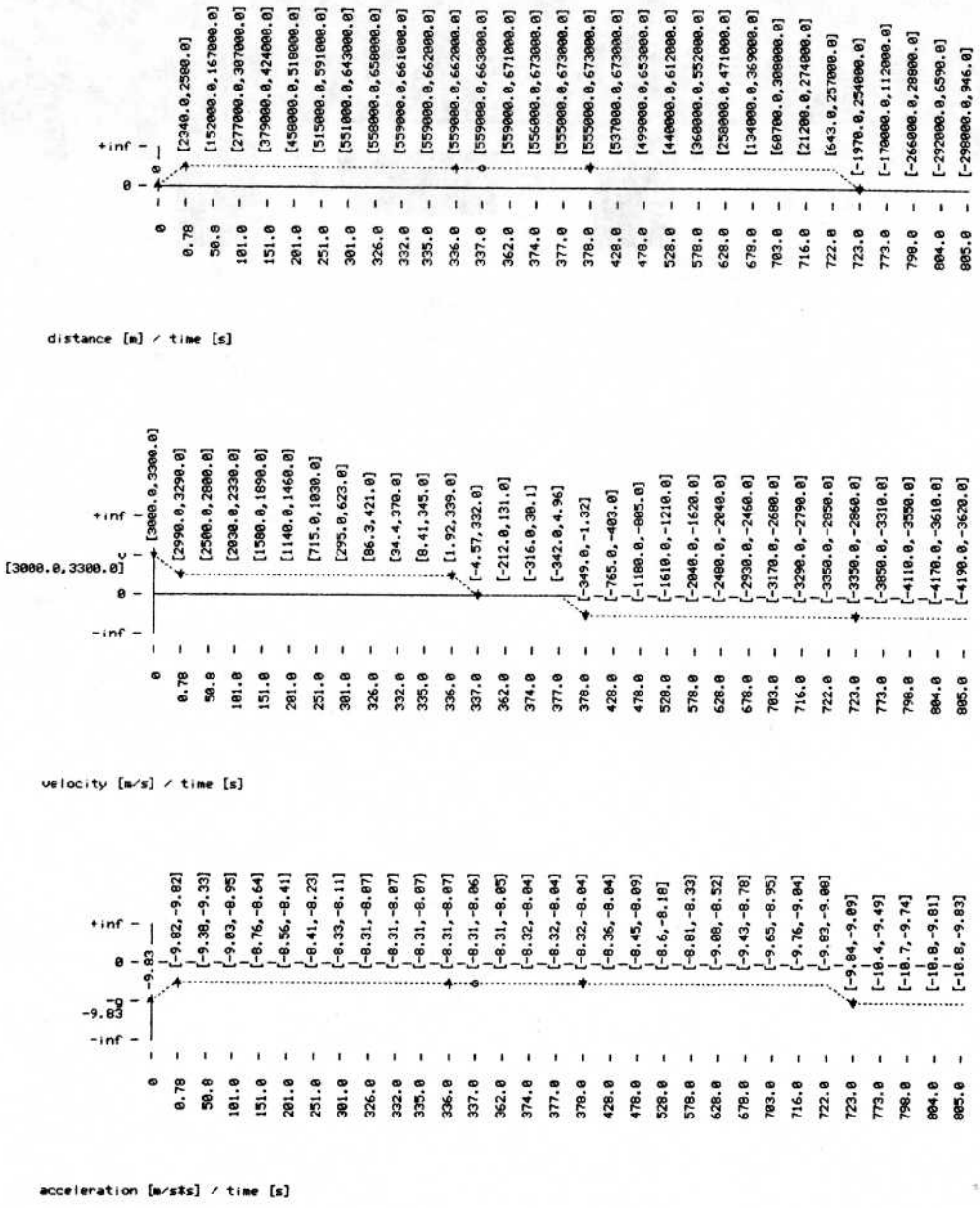


Figure 3: The plot shows the only possible behaviour that results from a simulation run using Euler's method. At the time points where the integration method reaches a landmark we can observe that the time steps get closer. The qualitative predicted time points $t_{q0}$, $t_{q1}$, $t_{q2}$, $t_{q3}$, $t_{q4}$ and $t_{q5}$; to be found at $0s$, $0.78s$, $336s$, $337s$, $378s$ and $723s$. They are indicated by the symbols ↑,↑,↑, ∘, ↓ and ↓, e.g. for distance and acceleration. The connecting slope in the plots has no other meaning than visual guidance.

| | Euler | | RK4 | Q3 | | Control |
|---|---|---|---|---|---|---|
| $t_1[s]$ | $[336, 378]$ | $[337, 379]$ | $[338, 379]$ | $[334, 384]$ | $[335, 382]$ | $[339, 381]$ |
| $t_2[s]$ | $[722, 805]$ | $[693, 777]$ | $[675, 758]$ | $[607, 915]$ | $[607, 835]$ | $[677, 761]$ |
| $max_D[km]$ | $[559, 673]$ | $[518, 633]$ | $[493, 607]$ | $[457, 666]$ | $[470, 646]$ | $[495, 609]$ |
| # states | 25 | 51 | 23 | 25 | 50 | 7614 |

Table 1: From the plots we can observe that the rocket reaches the maximum height at $t_1$ and it turns back on earth at $t_2$. The maximum distance from earth is given by $max_D$. The table shows three significant digits of the results. The fourth row holds information about the number of interpolated states. Two simulation runs have been performed using Euler's method and one using Runge Kutta $4^{th}$ order (RK4). For comparison the results of Q3 and an iterative summation method with step size $0.1s$ as reference (Control) are shown in the last three columns.

ues. Although Runge Kutta $4^{th}$ order constantly delivers by approximately two seconds smaller time interval values for the boundaries compared to the reference values, it is closest to the control values of all methods examined. The only exception is the upper bound predicted by Q3 for $t_1$ which is slightly more accurate than Runge Kutta $4^{th}$ order's corresponding value. However, the lower boundary predicted by Q3 is much more far off than the prediction by Runge Kutta $4^{th}$ order and results in a wider and less precise interval.

Comparing the results using Euler's method and the Q3 method, we can observe that even with just 25 interpolated states our algorithm predicts tighter bounds for $t_1$, $t_2$ and $max_D$ then Q3 does with 50 interpolated states. This is due to the landmark approximation algorithm. Q3 produces weak results for the upper and lower boundaries which can be observed particularly for the prediction of $t_2$. Our method predicts $t_2$ to be $[722, 805]s$ while Q3 predicts $[607, 915]s$, for 25 interpolated states. The closest value predicted by Runge Kutta $4^{th}$ order is $[675, 758]s$ and the resulting interval smaller by 36% even for twice the number of interpolated states in Q3. The result of the simulation run using Euler's method with 25 interpolated states is shown in Figure 3, the qualitative predicted time points $t_{q0}$, $t_{q1}$, $t_{q2}$, $t_{q3}$, $t_{q4}$ and $t_{q5}$ are indicated by the symbols $\uparrow$, $\uparrow$, $\uparrow$, $\circ$, $\downarrow$ and $\downarrow$, e.g. for distance and acceleration. We can observe that the time intervals, which preceed qualitative time points are smaller sizes, i.e. they provide a refined linear approximation.

The first two time points in Figure 3 are at $t = 0s$ and $t = 0.78s$. These are identical with the first two qualitatively predicted time points $t_{q0}$ and $t_{q1}$. Here, the landmark approximation algorithm has been active for the first time. The parameter distance has the following values: $\texttt{rocket(dist)}(t_{q0}) = (0, 0, inc)$ and $\texttt{rocket(dist)}(t_{q1}) = ((0, +inf), inc)$. The numeric integration could not interpolate a state that matches with $qm(\texttt{rocket(dist)}(t_{q0})) = 0$, so the step size has been halved until it became less than $\varepsilon = 1s$. This is the case at $t = 0.78s$. With respect to Definition 4.1 the semi-quantitative values of this state have been propagated to the qualitative predicted state at $t_{q1}$.

The effect of curve correction can be observed in the interval $[t_{q1}, t_{q2}]$, at $[t_{0.78}, t_{336}]$. From $t = 301s$, the linear approximation produces results that lie outside the scope of a qualitative magnitude or that are incompatible with the direction of change of a parameter. In this case a newly interpolated state is accepted only if it is compatible with the parameter's qualitative values or the step size becomes less than $\varepsilon$.

From $t = 0.78s$ the algorithm proceeds in steps of $50s$. The next time point after $t = 301s$ would be $t = 351s$, but the algorithm detects a conflict in the direction of change: $sqm(\texttt{rocket(distance)}(t_{301})) > sqm(\texttt{rocket(distance)}(t_{351}))$ and $qdir(\texttt{rocket(distance)}(t_{301})) = inc$. The step size is halved and the parameter value at the newly computed time point $t = 326s$ is compatible with the direction of change. The algorithm proceeds in steps of $25s$, but the next time point would be $t = 351s$ and again this conflicts with $qdir(\texttt{rocket(distance)}(t_{326})) = inc$. This time the algorithm rejects the newly computed time point twice. The step size is $6.25s$ and a new state is interpolated at $t = 332s$. The step size is halved again to $3.125s$. A new state is accepted at time point $t = 335s$. The step size is halved twice and the algorithm stops, because $\varepsilon \geq 0.78125$. The magnitudes of the newly interpolated state $t = 336s$ are propagated to the qualitative predicted state $t_{q2}$, with respect to Definition 4.

The next qualitative interval $[t_{q2}, t_{q3}]$, at $[t_{336}, t_{337}]$, has no interpolated states. The step size has been halved until it has been less than or equal to $\varepsilon$. The magnitudes computed for $t = 337s$ have been propagated to $t_{q3}$.

At the time points $t = 0s$ we observe $sqm(\texttt{rocket(accel)}(t)) = -9.83\frac{m}{s^2}$. This magnitude has been computed initially and has been used to annotate the landmark `-g` in the quantity space of parameter `accel`. At $t = 773s$ the qualitative simulation predicted $\texttt{rocket(accel)}(t_{q5}) = (-g, dec)$. After some interpolated states, landmark approximation is applied and the last interpolated state is propagated to $t_{q5}$ at $t = 773s$ in Figure 3, with respect to Definition 4.2.c.

## Discussion

The work presented in this paper demonstrates that landmark approximation improves the prediction of numeric magnitudes and leads to tighter intervals at possibly interesting time points than numeric interpolation alone could do for the same step size. With the combination of all available symbolic and numeric information, we gain the precise information on the critical or interesting states of the system. This statement is additionally supported by the effect of curve correction because the numeric interpolation is controlled by a previously predicted qualitative behaviour. Also the representation of semi-quantitative values is an important key to the improved prediction. Although a landmark in a parameter's quantity space can be bound to a numeric magnitude, a predicted interval will only match if this magnitude lies within the predicted numeric interval.

The presented transition table does not provide transitions to or from intermediate states. We assume the direction of change of a parameter to be constant between two time points. The advantage of this method is that all changes in the parameter values are identifiable at particular time points. In systems like QSim (Kuipers 1994), that use two transition tables for point-to-interval and interval-to-point transitions, a parameter value can be steady at a time point and increasing at the next state, which is a time interval.

Although there exists some semi-quantitative extensions to QSim, these extensions have some weaknesses. Semi-quantitative predictions of Q2 (Kuipers & Berleant 1988) over time intervals can be very weak when the distance between time points becomes very large, because it does not interpolate states in time intervals. To overcome this weakness Q2 has been extended to Q3 (Berleant 1991; Berleant & Kuipers 1997). It uses an step size re-

finement algorithm to subdivide intervals predicted by Q2 and interpolates new time points between existing ones. In contrast to our work, Q3 can not reach a finer granularity at the time points than across intervals. Furthermore, both Q2 and Q3 rely on the mean value theorem for interval propagation, whereas we provide the use of Runge Kutta $2^{nd}$ order or Runge Kutta $4^{th}$ order and gain a higher accuracy for the semi-quantitative predictions.

NSim, *dynamic envelope simulation*, presented in (Kay 1996; Kay & Kuipers 1993) extends a QSim QDE to a *semi-quantitative differential equation*, SQDE. An extremal ordinary differential equation system is dervied from a SQDE by minimizing and maximizing each derivative equation. This system is simulated using numeric integration such as Runge Kutta. In contrast, our integration methods are capable to handle interval arithmetic as well. SQSim (Kay 1996; 1998) combines Q2 and NSim. Predictions of Q2 and NSim can be intersected and must be consistent, otherwise the behaviour can be refused. SQSim can make more precise predictions than each of the methods alone could do. However, SQSim inherits the limitations of QSim as discussed above.

Based on QPT (Forbus 1984), SimGen is presented in (Forbus & Falkenhainer 1990; 1992). SimGen performs a qualitative analysis on a qualitative model to predict possible qualitative behaviours. For each behaviour a numeric model can be inferred for a numeric simulation run. Pika (Amador, Finkelstein, & Weld 1993) is another self-explanatory simulator, similar to SimGen. In contrast to SimGen it performs much better and the modelling language allows arbitrary algebraic and differential equations. Like our simulation engine, the method used for numeric integration is Runge Kutta $4^{th}$ order with an adaptive step size control. However, in our approach predicted qualitative behaviours are controlled by semi-quantitative information and interpolated behaviours are controlled by qualitative information, where in contrast neither SimGen nor Pika use qualitative and semi-quantitative models for simulation.

SQPC (Farquhar & Brajnik 1995), is an extension of QPC (Crawford, Farquhar, & Kuipers 1990). SQPC models are translated into SQDEs, an extension of QDEs. The simulation algorithms used are Q2 and NSim which are extensions of QSim as discussed above. Similar to our approach, both QPC and SQPC combine the conceptual modelling mechanism of QPT and the mathematical methods representation within QSim, but in contrast we allow the direct definition of objects and processes by the use of qualitative or semi-quantitative constraints.

Fuzzy mathematics is introduced for qualitative modelling (Nordvik, Smets, & Magrez 1988), diagnosis (Vescovi & Robles 1992) and process engineering (Bousson & Travé-Massuyès 1993) as well as a semi-quantitative extension to qualitative process theory (D'Ambrosio 1989) and qualitative simulation (Shen & Leitch 1993) to resolve ambiguities that are inherent to qualitative approaches. Although not all spurious behaviours can be pruned fuzzy mathematics helps to reduce the number of spurious behaviours significantly. However, no numeric interpolation is provided using fuzzy mathematics.

*Numerical Interval Simulation*, NIS, presented in (Vescovi, Farquhar, & Iwasaki 1995) is based on fuzzy simulation. Interval arithmetic is used to compute the possible values of state variables. However, this system does not provide qualitative reasoning techniques to combine symbolic and semi-quantitative information.

## Conclusion

There are a number of systems that rely on qualitative reasoning techniques which can combine numeric and qualitative information.

In this report, we present a new methodology to increase the precision of semi-quantitative predictions at qualitative predicted time points that highlight the critical or interesting states of a system's behaviour. This methodology takes advantage of our transition table as well as the way of propagating semi-quantitative information. The results of the simulation runs have been evaluated in four ways. First, by comparison to simulations with various step sizes. Second, by comparison to the results of a separately implemented "C"-version of the integration methods. Third, by comparison to an iterative summation method. And finally, by comparison to one established semi-quantitative simulation tool.

The system is implemented to work as a semi-quantitative reasoning server, where multiple clients can connect and build simulation models from the connected molecular biology knowledge base and afterwards perform a simulation run (http://cogito.rz-berlin.mpg.de/~ biosim).

## References


Alefeld, G., and Herzberger, J. 1983. *Introduction to Interval Computation.* NY: Academic Press. Translated by J. Rokne from *Einführung in die Intervallrechnung.*

Amador, F. G.; Finkelstein, A.; and Weld, D. S. 1993. Real-time self-explanatory simulation. In *Proc. 11th National Conf. on Artificial Intelligence (AAAI-93)*, 562–567.

Berleant, D., and Kuipers, B. 1997. Qualitative and quantitative simulation: Bridging the gap. *Artificial Intelligence* 95(2):215–255.

Berleant, D. 1991. *The use of Partial Quantitative Information with Qualitative Reasoning.* Ph.D. Dissertation, Department of Computer Sciences, University of Texas at Austin, Austin, Texas.

Bousson, K., and Travé-Massuyès, L. 1993. Fuzzy causal simulation in process engineering. In *Proc. 13th Int. Joint Conf. on Artificial Intelligence (IJCAI-93)*, 1536 – 1541. San Mateo, CA: Morgan Kaufmann.

Crawford, J.; Farquhar, A.; and Kuipers, B. 1990. QPC: A compiler from physical models into qualitative differential equations. In *Proc. 8th National Conf. on Artificial Intelligence (AAAI-90)*, 365–372. San Mateo, CA: Morgan Kaufmann.

D'Ambrosio, B. 1989. Extending the mathematics in qualitative process theory. In Widman, L. E.; Loparo, K. A.; and Nielsen, N. R., eds., *Artificial Intelligence, Simulation, and Modeling.* New York: Wiley-Interscience. 133–158.

de Kleer, J., and Brown, J. S. 1984. Qualitative physics based on confluences. *Artificial Intelligence* 24:7–83. Also in *Readings in Knowledge Representation*, Brachman and Levesque, editors, Morgan Kaufmann, 1985, pages 88-126.

Farquhar, A., and Brajnik, G. 1995. A semi-quantitative physics compiler. In *Proc. 9th Int. Workshop on Qualitative Reasoning About Physical Systems.*

Forbus, K. D., and Falkenhainer, B. 1990. Self-explanatory simulations: An integration of qualitative and quantitative knowledge. In *Proc. 8th National Conf. on Artificial Intelligence (AAAI-90)*, 380–387. Menlo Park, CA: AAAI Press/The MIT Press.

Forbus, K. D., and Falkenhainer, B. 1992. Self-explanatory simulations: Scaling up to large models. In *Proc. 10th National Conf. on Artificial Intelligence (AAAI-92)*, 685–690. Menlo Park, CA: AAAI Press/The MIT Press.

Forbus, K. D. 1984. Qualitative process theory. *Artificial Intelligence* 24:85–168.

Heidtke, K. R., and Schulze-Kremer, S. 1998. Biosim - a new qualitative simulation environment for molecular biology. In Glasgow, J.; Littlejohn, T.; Major, F.; Lathrop, R.; Sankoff, D.; and Sensen, C., eds., *Proceedings of the 6th International Conference on Intelligent Systems for Molecular Biology*, 85–94. AAAI Press.

Hyvönen, E. 1992. Constraint reasoning based on interval arithmetic: the tolerance propagation approach. *Artificial Intelligence* 58:71 – 112.

Kay, H., and Kuipers, B. 1993. Numerical behavior envelopes for qualitative models. In *Proc. 11th National Conf. on Artificial Intelligence (AAAI-93)*, 606–613. Menlo Park, CA: AAAI Press/The MIT Press.

Kay, H. 1996. *Refining imprecise models and their behaviors.* Ph.D. Dissertation, Department of Computer Sciences, University of Texas at Austin, Austin, Texas.

Kay, H. 1998. SQsim: A simulator for imprecise ODE models. *Computers and Chemical Engineering* 23(1):27–46.

Kuipers, B. J., and Berleant, D. 1988. Using incomplete quantitative knowledge in qualitative reasoning. In *Proc. 7th National Conf. on Artificial Intelligence (AAAI-88)*, 324–329. San Mateo, CA: Morgan Kaufmann.

Kuipers, B. 1986. Qualitative simulation. *Artificial Intelligence* 29:289 – 338.

Kuipers, B. 1994. *Qualitative Reasoning. Modeling and Simulation with Incomplete Knowledge.* Cambridge, MA: MIT Press. http://www.cs.utexas.edu/users/qr/QR-software.html.

Nordvik, J.; Smets, P.; and Magrez, P. 1988. Fuzzy qualitative modeling. In Bouchon, B.; Saitta, L.; and Jager, R., eds., *Second International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, 231 – 238. Berlin: Springer-Verlag.

Shen, Q., and Leitch, R. 1993. Fuzzy qualitative simulation. *IEEE Trans. on Systems, Man and Cybernetics* 4(23):1038–1061.

Struss, P. 1990. Problems of interval-based qualitative reasoning. In Weld, D., and de Kleer, J., eds., *Qualitative Reasoning about Physical Systems.* Morgan Kaufmann. 288–305.

Vescovi, M., and Robles, J.-P. 1992. Fuzzy diagnosis of continuous processes. In Neumann, B., ed., *Proc. 10th Eur. Conf. on Artificial Intelligence*, 749 – 753. Vienna, Austria: John Wiley & Sons.

Vescovi, M.; Farquhar, A.; and Iwasaki, Y. 1995. Numeric interval simulation: Bounding behaviors of non-monotonic systems. In *Proc. 14th Int. Joint Conf. on Artificial Intelligence*, 626 – 633. San Mateo, CA: Morgan Kaufmann.