Behaviour Prioritisation in Fuzzy Qualitative Simulation.

George Macleod Coghill
Department of Computing Science, Unviersity of Aberdeen AB24 3UE, Scotland, UK.
e-mail: gcoghill@csd.abdn.ac.uk

Abstract

Fuzzy qualitative simulation combines the features of qualitative simulation and fuzzy reasoning in order to gain advantages from both. However, the output of a fuzzy qualitative simulation process is a behaviour tree which for complex systems will be large. In order to overcome this and permit focussing on preferred behaviours *priortisation* was developed. In this paper a new prioritisation scheme is presented that makes use of both constraint and temporal information to perform the prioritisation.

Keywords: Fuzzy Qualitative Reasoning, Prioritisation

1 Introduction

One of the original motivations for the development of Qualitative Reasoning (QR) systems was a research programme to enable expert systems to reason from first principles, in order to overcome perceived weaknesses inherent in the first generation, rule-based, expert systems [7]. QR gives a broad picture of the way in which a system can behave and it was not long before the engineering community became interested in, and contributed to the field, because it was seen as a useful tool for simulating the behaviour of complex but incompletely specified plant. These influences have contributed to the utilisation of semi-quantitative information [1].

On the other hand Fuzzy systems were also developed to overcome some limitations in rule-based systems, by extending them to handle approximate knowledge. However, whereas QR deals with incomplete structural models, Fuzzy systems have tended to deal with input/output type problems. This has not been exclusively the case though, and fuzzy sets have been combined with interval simulators to carry out fuzzy interval simulations [2]. However, as with normal interval simulation the goal has been to generate narrowly focussed *unique* behaviours.

This situation led to the development of systems that combined the features of qualitative reasoning with those of fuzzy systems [9, 3]. There are at least three advantages which ensue from the combination of fuzzy and qualitative approaches [9]:

- the fact that the meaning of a qualitative value and its support set (the real number line here) are captured in a single representation,
- the ability to incorporate empirical knowledge into a model (which is also finer grained than the $M^{+/-}$ constraint utilised in QSIM [7]), and
- being able to include more detailed knowledge of the temporal behaviour of the variables in a model than the total ordering available within QSIM, which is useful for use in such applications as model-based diagnosis and control.

This was the motivation behind the development of FuSim [9], which is the system which was the major influence on the development of *Morven*. However, QR systems, regardless of complexion, generate behaviours that are not unique; and in the case of complex systems may be prohibitively large. In order to ameliorate this steps have been taken to assign a priority to the behaviours in a fuzzy behaviour tree [8]. It is the analsysis and development of such prioritisation schemes that is the subject of this paper.

The structure of this paper is as follows. In the next section the *Morven*[1] toolset is summarised, in order to put the subject of this paper in its overall context. In section 2 the details of the original prioritisation scheme are presented

1. This system was previously known as *Mycroft*, but I discovered that this name was far from unique so I changed it to the name of my elder daughter.

and criticised; and this is followed (in section 4) by the description of an alternative and extended approach to behaviour prioritisation. In section 5 an illustrative example is described and analysed, and from this some relevant conclusions drawn.

2 The *Morven* Toolset

The *Morven* toolset is a constraint-based fuzzy qualitative reasoning system containing a number of simulation and envisionment algorithms. The development of this toolset has permitted the suitability of different techniques to be examined in a number of contexts; and the comparison of different approaches to constraint-based fuzzy qualitative simulation to be made [3].

2.1 Representation and Inference

Fuzzy sets extend the ideas of traditional set theory to include the concept of partial (or graded) membership. It is assumed here that the ideas underlying fuzzy sets are known to the reader; however, a description of the domain and explanation of the concepts may be found in [6]. In FuSim, for reasons of computational efficiency, trapezoidal fuzzy numbers and intervals are used. An example of such a fuzzy number, a parameterised four tuple, is shown in Figure 1. This fuzzy number meets the criteria for being a fuzzy set in that it has a graded membership. The set is described by its membership function, $\mu_a(x)$, which is described by the four tuple [a, b, α, β] and is defined as:

$$\mu_a(x) = \begin{cases} 0 & x < a - \alpha \\ \alpha^{-1}(x - a + \alpha) & x \in [a - \alpha \quad a] \\ 1 & x \in [a \quad b] \\ \beta^{-1}(b + \beta - x) & x \in [b \quad b + \beta] \\ 0 & x > b + \beta \end{cases}$$

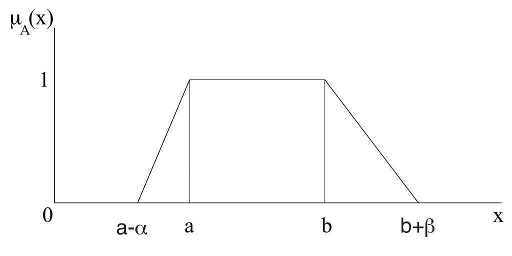

Figure 1: A Trapezoidal Fuzzy Number

The quantity space which is built from fuzzy numbers must be closed, continuous, finite and cover all values which a variable can take. An example of such a quantity space is shown in Figure 2. It is also possible to turn a trapeziodal fuzzy interval into a crisp interval by means of α-cuts [9]. Here a particular membership value (α) is chosen as representing "typicality" with respect to the fuzzy quantity, or quantity space, of interest; and then these 'typical' values can be used in the reasoning process as was done in FuSim, and carried over into *Morven*. Of course, what constitutes 'typical' can be altered by selecting different values for α. In fuzzy qualitative simulation, unlike QSIM, the quantity space for the derivatives of a variable may also be dense (that is, can have any number of divisions consistent with the definition of a quantity space).

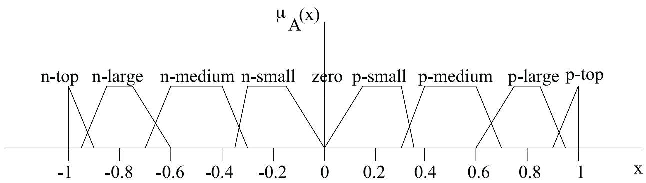

Figure 2: A Fuzzy Quantity Space

The *Morven* toolset is a qualitative reasoning system within the so called constraint based ontology. The models used in the toolset consist of sets of variables and the constraints that relate them. In fuzzy qualitative reasoning the operators utilised are the same as for its symbolic counterpart, though by the nature of the case there is a difference in the way they are implemented. All the variables of the system take their values from a predefined fuzzy quantity space.

In *Morven* the model constraints are causally ordered [5] and distributed over a number of differential planes [10]. That is, the qualitative differential equation (*qde*) model is developed on plane-0 (the zeroth differential plane), and the relationships between the higher derivatives of the system are obtained by differentiating the *qde* and representing the results as a *qde* on the so called higher differential planes.

Finally, as with QSIM and FuSim, *Morven* represents variables as a vector consisting of the fuzzy qualitative value for the magnitude and derivatives of the variable as a function of time. However, whereas QSIM and FuSim only make use of vectors of length 2 (representing the magnitude and direction of change of the variable), *Morven* can use vectors of any required length. For practical purposes three is the most that is generally required for exogenous variables, permitting a description of the *curvature* of the function. For any model of a state system there is a relationship between the number of differential planes in the model and the length of the vectors describing the state variables: for a model containing n differential planes, the state variable vector will contain $n+1$ elements. For example, in the single tank case above, the state variable, h, will require a vector of length 3:

$$h = [d^0 d^1 d^2]$$

where d^0, d^1 and d^2 are the zeroth, first and second derivatives of the variable respectively (or, if preferred: the magnitude, direction and curvature). When applied to an actual system variable, these elements are referred to as variable-vector elements

As stated previously, the *Morven* toolset consists of a number of simulation and envisionment algorithms [3, 4]. However, all these algorithms, in common with qualitative reasoning approaches in general, divide the inference process into two phases: Qualitative Analysis (QA) and Transition Analysis (TA).

In the QA phase the equations of the system model are solved and qualitative states generated. In the TA phase the values of the magnitudes and derivatives of, at least, the state variables of the system are known, and this information is used (along with transition rules) to decide which values these variables may take in the succeeding time interval (or time point).

2.2 Temporal Calculations

One of the motivations for combining qualitative reasoning with fuzzy reasoning was the provision of fuller temporal information to be included in the qualitative reasoning process. There are four temporal calculations that are important in *Morven*: the *persistence time*, the *relative arrival time*, the *absolute arrival time*, and the *absolute departure time*.

Persistence time, ΔT_p: This is a time interval representing the endpoint of the interval during which an element of a variable vector remains in the same state, assuming that the calculation is made from the time the variable entered that state, and the derivative used in its calculation does not change during the time interval. The persistence time for an element of a variable vector, d^n, is defined as:

$$\text{If } 0 \notin d_\alpha^{n+1}, \text{ then } \Delta T_p \in \frac{W(d^n)}{|d^{n+1}|_\alpha}$$

where d^n and d^{n+1} are the nth and $(n+1)$th derivatives of the variable (for the purposes of these calculations the magnitude of the variable is considered as the 0th derivative). $W(d^n)$ is the α-width of the fuzzy interval of the nth derivative. This formula is the same as that used in FuSim.

Relative Arrival time, ΔT_a: This is the time interval representing the length of time it takes for a variable-vector element to transit from one state to another. Consider the case where d_j^n and d_{j+1}^n are jth and $(j+1)$th quantities from the quantity spaces of the nth derivative of the variable under consideration (likewise, d_j^n and d_{j+1}^{n+1} are jth and $(j+1)$th quantities from the quantity spaces of the $(n+1)$th derivative of that variable.). Then the relative arrival time for an element of a variable vector, d^n, used in a *Morven* simulation is given as:

$$\text{If } 0 \notin d_\alpha^{n+1}, \text{ then } \Delta T_a \in \frac{L[d_{j+1}^n]_\alpha - U[d_j^n]_\alpha}{|d^{n+1}|_\alpha}$$

where, the $L[\cdot]_\alpha$ and $U[\cdot]_\alpha$ are the lower and upper bounds of the respective α-intervals. If the nth element does not transit then the relative arrival time is zero. However, depending on whether the $(n+1)$th element transits, the value of $|d^{n+1}|_\alpha$ will be different. On the assumption that the present value of the $(n+1)$th element is d_j^{n+1}, if this element does not transit then $|d^{n+1}|_\alpha = |d_j^{n+1}|_\alpha$. However, if there is a transition then:

$$|d^{n+1}|_\alpha = |d_{j+1}^{n+1}|_\alpha - |d_j^{n+1}|_\alpha$$

Absolute Arrival time, T_A: This is the time interval representing the length of time it takes a variable vector element, d^n, to arrive at a particular state from the initial time of the simulation ($t = 0$). The formula for this is:

$$T_{A_n} = \sum_{i=0}^{i-1} \Delta T_{p_i}(d^n) + \sum_{j=1}^{n} \Delta T_{a_j}(d^n)$$

where n is the nth absolute time index.

Absolute Departure time, T_D: This is the time interval representing the length of time it takes a variable vector element to depart from a particular state, with the initial time of the simulation as the datum. The formula for the absolute departure time is:

$$T_{D_n} = T_{A_n} + \Delta T_{p_n}$$

There is an exception to this formula if the absolute time contains a transition of the (n+1)th element of the variable-vector. The expression utilised in such cases is:

$$T_{D_{n+1}} = T_{D_n} \Xi \Delta T_{p_n} + \Delta T_{p_{n,n+1}}$$

where, Ξ is a retraction operator, and $\Delta T_{p_{n,n+1}}$ is a pseudo-persistence time in which the $(n+1)$th element used in the calculation is the difference between the two quantities appearing in the transition of the $(n+1)$th element.

3 Prioritisation

Qualitative simulation does not result in a unique behaviour; rather it generates a tree of behaviours. In symbollic qualitative simulation the maximum branching factor is 4, whereas in fuzzy qualitative simulation it is 6. This means that for a complex system, even ignoring the problem of spurious behaviour generation, the behaviour tree may be large.

In response to these results Leitch and Shen [8] developed a scheme for prioritising the states and behaviours generated in a simulation, on the basis of a distance metric, in order to find the best plausible approximate behaviour. In this section the details of the particular distance metric utilised by FuSim is given. This is followed by a description of the method used to prioritise the states generated by FuSim. Finally the technique for prioritising the behaviours in a FuSim behaviour tree is assessed.

3.1 A Distance Metric

To understand the distance metric introduced by Leitch & Shen (which is also utilised by *Morven*) and how it is used, it is necessary to give a more detailed example of two kinds of value which are used in a simulation: the *predicted value* and the *propagated value*. The former is the value (or set of values) which a variable is assigned on the basis of the transition rules; that is they are the values predicted by the integration phase of the simulation. Each system constraint consists of a constrained variable, which is the variable appearing on the left hand side of a constraint, and one or more constraining variables which appear on the right hand side of a constraint and may be used to calculate a value for the constrained variable. The propagated value of a variable is the value thus calculated.

Consider a system consisting of the following three place constraint:

$$a = b + c$$

If the constraining variables b and c have the α-cut interval values [1 4] and [5 8] respectively, then the propagated value for a will be [6 12], as shown in Figure 3.

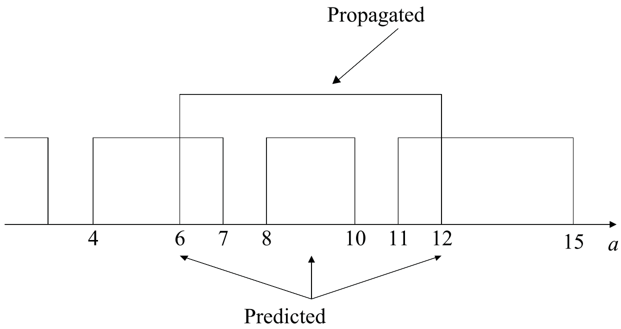

Figure 3: Propagated and predicted values

It is possible that the predicted and propagated values are identical; however, it is usually the case that a propagated value will intersect with several predicted values, as shown in Figure 3. One can treat each predicted value that intersects with the propagated value as equally possible. However, since not all the members of a variables quantity space will intersect the propagated value to the same degree the method of prioritisation was developed to reflect this fact and give those quantities which are a closer approximation to the propagated value a higher priority. The measure of which quantity is the best approximation is gained by means of the distance metric.

This distance metric (given the symbol d) is really a measure of similarity between two fuzzy numbers. Let the propagated value be depicted by a normal capital letter and the predicted value by a capital letter with a circumflex above it; then the formula for the distance between the two values given in [8] is:

$$d(A, \hat{A}) = [(power(A) - power(\hat{A}))^2 + (centre(A) - centre(\hat{A}))^2]^{\frac{1}{2}}$$

where, for 4-tuple parametric fuzzy numbers:

$$power([a, b, \alpha, \beta]) = \frac{1}{2}[2(b-a) + \alpha + \beta]$$
$$centre([a, b, \alpha, \beta]) = \frac{1}{2}[b + a]$$

These formulae represent the area of the fuzzy interval and the centre of the nucleus of the fuzzy interval respectively. If two fuzzy intervals are identical, then according to the above expressions they will have a distance of zero. Thus, the smaller the distance between a propagated and predicted value, the better the approximation.

As an example, consider again the situation

depicted in Figure 3. Call the propagated value $\hat{a}$, and the predicted values $a1, a2$ and $a3$ with values [4 7], [8 10] and [11 15] respectively. Then
$d(a1, \hat{a}) = [(3-6)^2 + (5.5-9)^2]^{0.5} = 4.6$
$d(a2, \hat{a}) = [(2-6)^2 + (9-9)^2]^{0.5} = 2$
$d(a3, \hat{a}) = [(4-6)^2 + (13-9)^2]^{0.5} = 5.7$
from which it can be seen that the value which must be assigned the top priority is $a2$, (followed by $a1$ and finally $a3$).

3.2 State Prioritisation

A system would not normally consist of only one constraint; therefore the states which are to be prioritised for any step ahead in the simulation will be made up of values for a number of different variables which are consistent with several different constraints in the system. Thus a method of prioritisation is required which will order complete system states. The approach suggested by Shen and Leitch deals with this task in two stages. The first stage provides the distance for a complete predicted variable value consisting of a magnitude and derivative $< A, B >$ from the equivalent propagated state $< \hat{A}, \hat{B} >$. Each element of the variable will have an associated distance $d(.)$; the distance for the complete variables, $D(.)$, is:

$$D(< A, B >< \hat{A}, \hat{B} >) = max\{d(A, \hat{A}), d(B, \hat{B})\}$$

The second stage is to find a distance for each complete system state from the distances for each complete variable. In their paper, Shen and Leitch suggest the following formula:

Prioritise the states such that $\rho((A_i, B_i)) = j$, $i = 1, 2, \ldots, M$, if

$$D_i = min\{\{D_k | k = 1, 2, \ldots, M\} - \{D_k | k < j\}\}$$

That is: for each constrained variable in the constraints, the constraint is applied to both the magnitude and derivative of the variable, and hence a distance between propagated and predicted values can be found for both the magnitude and derivative of the variable. The overall distance for the variable is then taken to be the maximum of these distances. Having obtained distances for each variable value in the state, the distance for the particular state is taken to be the minimum of these distances. Then the states are prioritised in accordance with these distances, from minimum to maximum.

The above approach provides an ordering of the states at each step in the simulation. However, this method effectively makes a single constraint responsible for the priority assigned to the state. This is because the magnitude and derivative distances calculated for a single pair are assigned from the application of a single constraint, and the maximum chosen. Then the ordering is performed by selecting the minimum distance for these maxima, which is the same as selecting the constraint that produced this *maxmin* value. This approach then, while providing an excellent start in the application of prioritisation to fuzzy qualitative simulation, does not utilise most of the information available about the model structure. Therefore an alternative approach to the prioritisation of the system behaviours is discussed in Section 4.

3.3 Behaviour Prioritisation

The prioritisation dealt with in the preceding section is associated with the QA phase of the simulation. This is the analysis which deals with the values of states at instants in time. However, what is important about simulations are the behaviours; therefore it is essential that the behaviours generated by the qualitative simulator be prioritised, rather than the states. Leitch and Shen also address this problem. However, their solution is to examine the distances and priorities of the states at each step of the simulation and base the estimate of which behaviour should be top priority on the combination of the distances calculated at each step. This amounts to being a depth first search through the tree, seeking to find a path that minimises some cost. Again, this solution can be criticised methodologically. It implicitly assumes that the QA phase is the only important part of the simulation and that there is no cost (or at least equal cost) in transiting from one state to another. In contrast to this it can be argued that since simulation comprises a TA phase as well as a QA phase, behaviour prioritisation should be based on a combination of the constraint prioritisation and a temporal prioritiser which can estimate the most likely transition.

4 An Alternative Approach

On the basis of the distances calculated between the predicted and propagated values, the predicted values can each be assigned a priority. For every arithmetic constraint there will therefore be (at least) one predicted value which has the top priority. Since the model consists of a conjunction of constraints the top priority state will be the one consisting of the conjunction of the top priority predicted value for each constraint.

4.1 Constraint Prioritisation

In *Morven* variables and models have complementary representations: variables are represented as vectors that are qualitative representations of Taylor series and the models exist in differential planes. In a Taylor series the higher derivatives have less weight in the calculation of the variables value at a future time; likewise in the present situation the higher derivatives carry less weight in the calculation of the priority of a state. Also, since *Morven* models tend to be causally ordered for a simulation the values predicted from constraints later in the list are dependant on the predicted values of variable-vector elements calculated earlier in the constraints list. Thus, the state priority decreases as the priorities of the predicted values of the variable-vector elements decrease, from the lowest derivative upward to the magnitude and from variable-vector elements calculated later to those calculated earlier in the constraints. For example, the second priority state will be the one with every predicted value having top priority, except the constrained variable of the final constraint of the system which will have second top priority. Obviously then, the lowest priority state is the one in which all the predicted values are of the lowest priority. To clarify this, consider the following pseudo example. A causally ordered model consists of three constraints:

$a = f(x)$, $b = f(a)$, $c = f(b)$

x is given and a, b and c are the variables which need to be calculated. After application of the constraints the variables have the following values:

$$a = \{qa1{:}1,\ qa2{:}2\}$$
$$b = \{qb1{:}2,\ qb2{:}1\}$$
$$c = \{qc1{:}2,\ qc2{:}1\}$$

where the qs are quantities from the appropriate quantity space and the numerals after the colons are the priorities assigned to the quantities by the constraint. Thus the eight states created with these values will be prioritised as in the order of the following list (from highest priority to lowest): $\{(qa1, qb2, qc2), (qa1, qb2, qc1), (qa1, qb1, qc2),$ $(qa1, qb1, qc1), (qa2, qb2, qc2), (qa2, qb2, qc1),$ $(qa2, qb1, qc2), (qa2, qb1, qc1)\}$
which if looked at as a sets of conjunctions of priorities would have the following form: $\{(1 \wedge 1 \wedge 1), (1 \wedge 1 \wedge 2), (1 \wedge 2 \wedge 1), (1 \wedge 2 \wedge 2),$ $(2 \wedge 1 \wedge 1), (2 \wedge 1 \wedge 2), (2 \wedge 2 \wedge 1), (2 \wedge 2 \wedge 2)\}$
which gives a total ordering of the priorities for a given model.

4.2 Temporal Prioritisation

In Section 2.2 the different time calculations performed by *Morven* were described, and two of them - the *absolute departure time* and the *absolute arrival time* are relevant here since it is the ongoing simulation that is being dealt with. These times give a measure of the time elapsed since the beginning of the simulation till the variable-vector element either departs from its present state, or arrives in its next state. These times are both intervals representing the earliest and latest times a departure or arrival could take place. The possible transitions that take place may be ordered, and thus prioritised, on the basis of these absolute times, from fastest to slowest.

By combining constraint and temporal prioritisation one obtains an ordering of the behaviours in the behaviour tree: the behavioural prioritisation

5 An Illustrative Example

In this section an example is presented to illustrate the concepts discussed in the previous sections. The coupled tanks system is chosen because it is complex enough to explicate the concepts and simple enough to be understood and analysed. A schematic diagram of the coupled tanks system is shown in Figure 4.

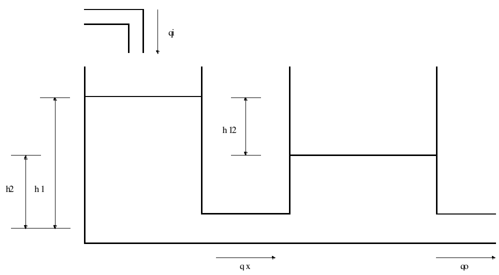

Figure 4: A Coupled Tanks System

Fuzzy qualitative reasoning permits the incorporation of empirical knowledge in a model in the form of fuzzy rules. The exact form in this example was chosen to emphasise the advantages of fuzzy qualitative reasoning in this respect, whilst keeping the problem tractable for ease of analsysis and explanation. The constitutive relationships of the coupled tanks system are fuzzy rule-bases of the following form:

$q_x = F(\Delta h)$	nt	nl	nm	ns	z	ps	pm	pl	pt
nt	1	0	0	0	0	0	0	0	0
nl	0	1	0	0	0	0	0	0	0
nm	0	0	1	0	0	0	0	0	0
ns	0	0	0	1	0	0	0	0	0
z	0	0	0	0	1	0	0	0	0
ps	0	0	0	0	0	1	0	0	0
pm	0	0	0	0	0	0	1	0	0
pl	0	0	0	0	0	0	0	1	0
pt	0	0	0	0	0	0	0	0	1

$q_o = F(h_2)$	nt	nl	nm	ns	z	ps	pm	pl	pt
nt	1	0	0	0	0	0	0	0	0
nl	0	1	0	0	0	0	0	0	0
nm	0	0	1	0	0	0	0	0	0
ns	0	0	0	1	0	0	0	0	0
z	0	0	0	0	1	0	0	0	0
ps	0	0	0	0	0	1	0	0	0
pm	0	0	0	0	0	0	1	0	0
pl	0	0	0	0	0	0	0	1	0
pt	0	0	0	0	0	0	0	0	1

and the structural relations are:

$$\Delta h = h_1 - h_2 \quad h_1' = q_i - q_x \quad h_2' = q_x - q_o$$

where h_1 and h_2 are the head of fluid in the tanks, Δh is the head difference, q_i is the rate fluid flow into the system, q_o is the flow of fluid from the system and q_x is the crossflow of fluid between the tanks.

These constraints are causally ordered for use with *Morven*. They also constitute the constraints of the zeroth differential plane; the constraints for the higher differential planes are obtained by differentiating these constraints. Of course, empirically derived rule bases cannot be differentiated and so this will only be achievable if empirical information has been obtained for the first differential plane as well. For the purpose of this example it is assumed that this has been done.[2]

5.1 Results and Discussion

In order to run a simulation, two pieces of information are required: the initial values of the states variables (h_1 and h_2), and a complete specification of the input (or exogenous) variable, q_i in this case. Consider the situation where there is a continuous steady flow into the tanks (which are initially empty) with value $p - medium$; this gives the following input description and initial values.

Input: $q_i =< p - medium\ zero >$

$$\text{Initial Values:} \quad \begin{cases} h_1 = zero \\ h_2 = zero \end{cases}$$

The result of this experiment is a behaviour tree containing 550 states. From this, one can select a number of paths to steady state that serve to illustrate the features of behaviour prioritisation. Three paths will be examined. A behaviour tree is also known as a partial envisionemnt because it constitutes that part of a complete envisionment graph reachable from a given initial starting state. In a behaviour tree the same state can appear in a number of different branches with a different state number and this can make it harder to discern what is going on when analysing the effects of prioritisation. In order to make things clearer, the same states for the three paths have been given the same label and the part envisionment graph is shown in Figure 5.

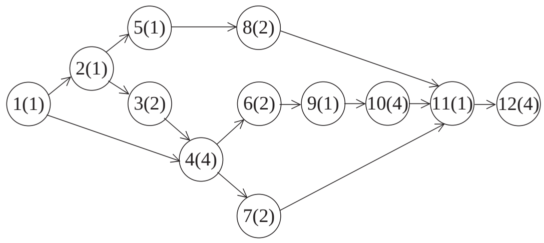

Figure 5: Partial behaviour graph with constraint prioritisation applied

This part envisionment starts with both tanks empty (state 1) and ends at an equilibrium state (state 12). Three states are common to all three paths (states 1, 11 and 12). The numbers in parentheses on the graph refer to the constraint priority of the states.

In order to demonstrate the effectiveness of the prioritisation scheme, the three paths chosen are: the fastest path (containing states: 1, 2, 5, 8, 11 and 12) taking a maximum time of 12.17 time units (tu); the shortest path (containing states: 1, 4, 7, 11, 12) taking a maximum time of 12.83tu; and a long path (containing states: 1, 2 3, 4, 6, 9, 10, 11 and 12) with a maximum time of 13.83tu.

Several things emerge perusal of Figure 5:

- The fastest path is also the one with the highest overall constraint priority. The exception to this is priority of state 8. However the top priority states at that level did not form any path to equilibrium and may therefore part of a spurious behaviour.
- The shortest path in terms of the number of qualitative states traversed is not necessarily the fastest. This can occur because some states persist longer than others; therefore, paths including them will be temporally longer than the other paths.
- Temporal prioritisation must be done globally rather than locally. This arises from

2. *Morven* can operate using only the zeroth differential plane, but for purposes of comparison two planes are used for this example.

the fact that the transition from state 2 to state 3 is faster than the transition state 2 to state 5; yet the former is part of the longest path and the latter is part of the fastest path overall. This is related to the problem of spurious behaviour generation, where it is well recognised that one of the sources of this problem is the local nature of the TA phase.

- The existence of different rational paths to equilibrium indicate that prioritisation is a means of behaviour selection. The constitutive relations in a fuzzy system contain vagueness and depending on the degree may allow the incorporation of a wide range of underlying parameter values. Here prioritisation acts as an analogue to sampling, enabling one to select either faster or slower changing behaviours as required.

Finally, on the assumption that an appropriate prioritisation scheme can be developed, the question arises as to how it should be integrated with existing filters. The process of prioritisation orders a set according to a metric. It does not actually allow one to state that any of the states should be eliminated. Implicit in this is the assumption that all the members of the set being prioritised are real. This leads to the conclusion that the existing methods for eliminating spurious behaviours [7] should still be used to eliminate as many behaviours as possible, and then (on the assumption that the remaining states are valid) prioritisation should be applied to select the most likely.

6 Conclusions

The theme of this paper has been the exploration of issues relating to prioritisation of the behaviours generated by fuzzy qualitative simulation. The possibility of selecting behaviours in this way was first suggested by Leitch and Shen [8], and they provided a useful initial method. However, their approach was based on *max* and *min* operators and did not utilize all the information available in making the decisions regarding which behaviour should be considered to have top priority. Therefore a new approach to prioritisation is suggested based on the recognition that constraint based models are conjunctions of constraints and the vectors representing the values of variables and qualitative versions of Taylor series. The resulting behavioiur prioritisation utilises both constraint and temporal prioritisation and is therefore a more versatile and informative version of prioritisation.

The conclusions from the experimental analysis of using this new prioritisation scheme are:

- The fastest path is also the one with the highest overall constraint priority.
- The shortest path may not be the fastest.
- Temporal prioritisation must be done globally rather than locally.
- The existence of different rational paths to equilibrium indicate that prioritisation is a qualitative analogue to sampling.

References

[1] Berleant D. & Kuipers B. J. 1997. “Qualitative and Quantitative Simulation: Bridging the Gap.” *Artificial Intelligence* 75: 215–255.

[2] Bonarini A. & Bontempi G., “A qualitative simulation approach for fuzzy dynamical models”, ACM Transactions on Modeling and Computer Simulation, 4(4), 285-313, 1994.

[3] Coghill G. M. “*Mycroft:* a Framework for Constraint Based Fuzzy Qualitative Reasoning”, Ph.D. Thesis, Heriot-Watt University, September 1996.

[4] Coghill G. M. “On the development of Fuzzy Envisionment” Proceedings of the 3rd International Workshop on Hybrid Methods for Adaptive Systems, Oulu Finland, July 10-11, 2003.

[5] Iwasaki Y., “Model based reasoning of device behavior with causal ordering”, Ph.D. Thesis, Carnegie Mellon University, August 1988.

[6] Kosko B., *Fuzzy Engineering*, Prentice-Hall International, 1997.

[7] Kuipers B. J., *Qualitative Reasoning: Modelling and Simulation with Incomplete Knowledge*, MIT Press, London, 1994.

[8] Leitch R. R. & Shen Q., “Prioritising behaviours in qualitative simulation”, in Proceedings of the 13th International Joint Conference on Artificial Intelligence, Pp 1523-1528, Chambercy, France, 28th August - 3rd September, 1993.

[9] Shen Q. & Leitch R. R., “Fuzzy qualitative simulation”, IEEE Transactions on Systems, Man and Cybernetics, 23, 1038 - 1061, 1993.

[10] Wiegand M. E., “Constructive Qualitative Simulation of Continuous Dynamic Systems”, Ph.D. Thesis, Heriot-Watt University, May 1991.