A Qualitative Model for Hybrid Control

Wolfgang Kleissl, Michael W. Hofbaur
Graz University of Technology
Institute of Automation and Control
Inffeldgasse 16c/2, A-8010 Graz, Austria
kleissl@irt.tu-graz.ac.at, hofbaur@irt.tu-graz.ac.at

Abstract

Complex artifacts that exhibit a mixture of continuous and discrete behaviors are difficult to control since one has to calculate a continuous control actuation that is interleaved with commanded discrete mode changes. Our paper presents a novel scheme for such a *hybrid control* problem that uses a qualitative description of the system's hybrid dynamics to pre-select feasible mode sequences prior pursuing the computationally expensive task of numerical control-law synthesis. We present a qualitative modeling framework that captures the hybrid model in terms of qualitative abstractions of a system's continuous behavior together with the description of the discrete dynamics (mode changes) and other control-induced constraints. The model's representation builds upon a structural analysis of the artifact's hybrid model and compactly encodes the hybrid dynamics as an Ordered Binary Decision Diagram like representation that supports fast on-line reasoning.

1 Introduction

Many modern systems exhibit hybrid behaviors that can be described in terms of continuous dynamics and discrete evolutions among operational modes. Automatic control of such systems involves the deduction of a hybrid control sequence that consists of a suitable mode sequence and the associated continuously valued actuation to the system. The deduction of this hybrid control strategy is difficult as the number of possible mode sequences over time can be very large.

A natural approach to tackle this problem is to capture the continuous dynamics of such a hybrid system in terms of a *qualitative model*, since then both, continuous and discrete dynamics of that system are described by relations among discretely-valued variables and one can apply symbolic methods from the toolkit of Artificial Intelligence to select among the possible controls. This paper follows this idea and presents an approach for hybrid control that builds upon a qualitative model to pre-select suitable control candidates. These candidates specify the mode-sequence but capture the continuous part of the hybrid control problem only qualitatively. An additional numerical optimization is then applied to determine the continuous control in more detail. Due to the qualitative pre-selection, we avoid that the computationally expensive numerical optimization is applied unnecessarily often.

Our qualitative model is formulated with its application, the task of on-line hybrid control, in mind. It represents the possible qualitative trajectories as a graph that enables efficient search methods to find a suitable hybrid control law. The graph itself is encoded similarly to an Ordered Binary Decision Diagram (OBDD) [Andersen, 1997], thus enables efficient search and compact storage.

In literature, qualitative methods in hybrid control mainly address issues of reconfiguration [Askari *et al.*, 1999] or fault diagnosis [Hamscher *et al.*, 1992]. Continuous control of hybrid systems, i.e. determination of continuous actuation, is preferably addressed by numerical methods, like mixed integer linear and quadratic programming [Bemporad and Morari, 1999]. Contrarily, we take a mixed qualitative/numerical approach that combines the insight provided by a qualitative approach with the exact solutions of a numerical approach. In this way we solve both tasks, mode configuration and continuous actuation. Together with the hybrid estimation of [Hofbaur and Williams, 2004; Hofbaur, 2005], in its 'final version', our approach to hybrid control shall address the problem of hybrid automation. This demands high flexibility with respect to reacting on unforeseen situations properly and motivates our aim to deduce hybrid control on-line.

The remainder of this paper is organized as follows: Section 2 addresses hybrid systems and introduces an example that guides through the article. Section 3 presents the problem of hybrid control and indicates how this can be tackled by qualitative pre-selection of promising control candidates. Section 4 introduces our qualitative modeling approach and discusses important features of the model such as efficient search within the model and compact compilation. Section 5 concludes this with a summary of compiling the example into a qualitative model.

2 Hybrid System

Let us start with a simple hybrid system that exhibits different dynamic behaviors according to its *operational mode* m_i. The dynamic system can be captured in terms of its continuously valued state $\mathbf{x}(t) = [x_1(t), \ldots, x_{n_x}(t)]^T$ and receives the inputs $\mathbf{u}(t) = [u_1(t), \ldots, u_{n_u}(t)]^T$. Its operation can be observed through the outputs $\mathbf{y}(t) = [y_1(t), \ldots, y_{n_y}(t)]^T$ and is described in terms of the linear model

$$\begin{aligned} \frac{d}{dt}\mathbf{x}(t) = &\quad \mathbf{A}_i\, \mathbf{x}(t) + \mathbf{B}_i\, \mathbf{u}(t) \\ \mathbf{y}(t) = &\quad \mathbf{C}_i\, \mathbf{x}(t) + \mathbf{D}_i\, \mathbf{u}(t)\ , \end{aligned} \tag{1}$$

where the index i of the system matrices $\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i$ and $\mathbf{D}_i$ indicates the dependence upon an operational mode m_i. Complexity of the system's operation is due to the hybrid evolution that consists of the dynamical change of the *continuous state* $\mathbf{x}(t)$ interleaved with discrete mode changes. The mode or *discrete state* $x_d(t)$ determines the dynamic model (1) that is valid at a particular time point t. Thus, the combination of continuous and discrete state

$$\mathbf{x}_h(t) = \langle x_d(t),\ \mathbf{x}(t) \rangle \tag{2}$$

manifests the *hybrid state* of the hybrid system.

As an example consider the hybrid system with two modes $\{m_1, m_2\}$. The variation of the dynamic behavior is due to different system matrices $\mathbf{A}_i$

$$\begin{aligned} \mathbf{A}_1 &= \begin{bmatrix} -0.35 & -1 \\ 1 & -0.35 \end{bmatrix}, \\ \mathbf{A}_2 &= \begin{bmatrix} 0.65 & 1 \\ -1 & 0.65 \end{bmatrix}. \end{aligned} \tag{3}$$

The input, output and direct transmission matrix are mode independent, more specifically[1]:

$$\mathbf{B}_i = \begin{bmatrix} 0.2 \\ 0 \end{bmatrix},\ \mathbf{C}_i = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},\ \mathbf{D}_i = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{4}$$

As in many real systems, we assume limited actuation, in particular, for this example we assume:

$$0 \leq \mathbf{u}(t) \leq 1\ . \tag{5}$$

Furthermore, we require that the continuous state $\mathbf{x}(t) = [x_1(t), x_2(t)]^T$ stays within a region of the state space such that

$$-1 \leq x_1(t) \leq 1,\ \ -1 \leq x_2(t) \leq 1\ . \tag{6}$$

These two constraints make the task of control difficult.

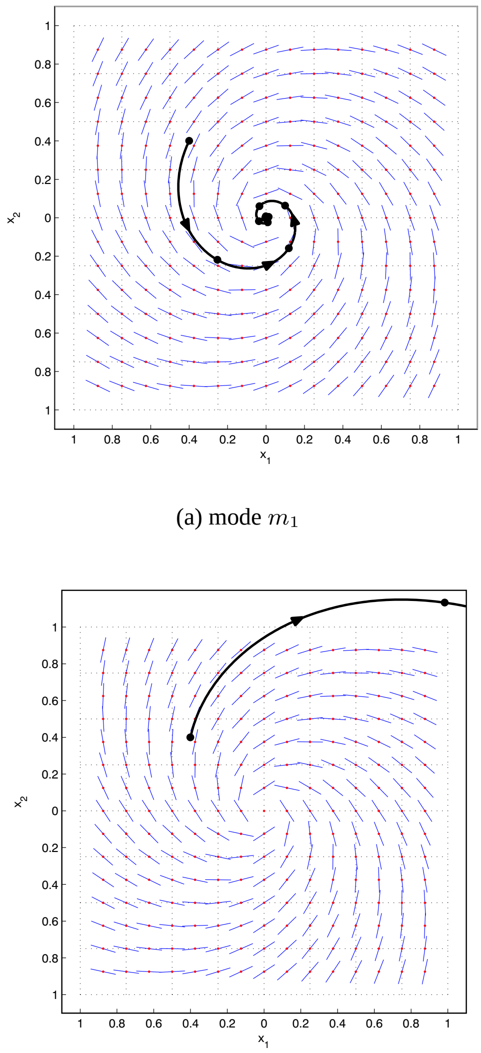
For simplicity, we assume here that mode transitions are purely triggered through an exogenous command input $\mathbf{u}_d$ at specified time points

$$t_k = t_0 + kT_s$$

where t_0 and T_s denote the initial time and the sampling period of our hybrid control system, respectively. Because of the *discrete-time* operation of our hybrid controller, we also assume that the value of the continuous input $\mathbf{u}(t)$ is kept constant within the sampling period at

$$\mathbf{u}(t) = \mathbf{u}_k, \quad t_k \leq t < t_{k+1}$$

and $\mathbf{u}_k$ denotes the discrete-time variant of the continuously-valued system input. In the same sense, $\mathbf{x}_k = \mathbf{x}(t_k)$ and



(a) mode m_1

(b) mode m_2

Figure 1: Vector fields and trajectories for the hybrid system

$\mathbf{y}_k = \mathbf{y}(t_k)$ denote the sampled continuous state and the system output, respectively. In terms of the discrete mode $x_d(t_k)$, we take the standard hybrid system's assumption that a mode change takes place immediately after the sampling time point t_k so that:

$$x_d(t) = x_{d\,k+1} \quad t_k < t \leq t_{k+1}\ .$$

Because of the hybrid controller's discrete time operation we model the hybrid system in terms of an automaton with two modes m_1 and m_2 that changes its discrete state immediately after t_k and specify the continuously-valued dynamics

[1] Although the example has a scalar input we will use the bold face vector notation $\mathbf{u} = [u]$ to denote the continuous input in order to be conform with the dynamic model (1).

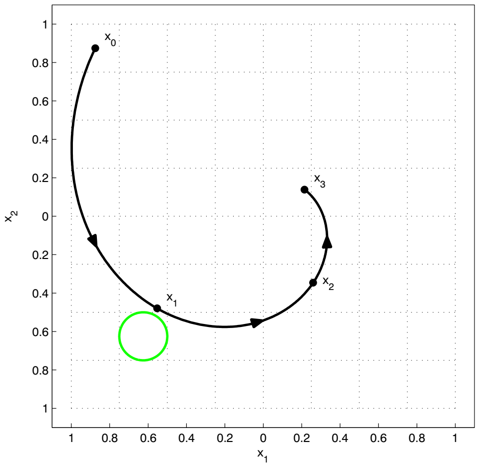


Figure 2: Trajectory for mode sequence $\{m_1, m_1, m_1\}$

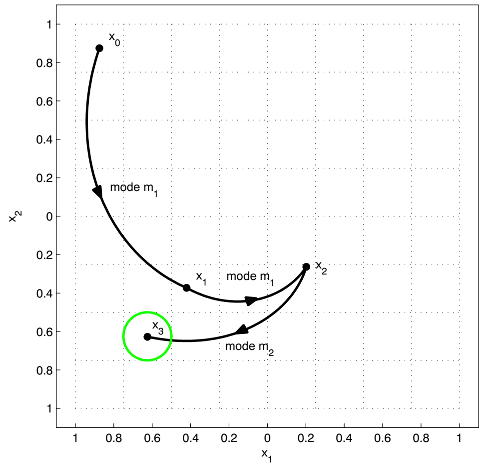


Figure 3: Trajectory for mode sequence $\{m_1, m_1, m_2\}$

in terms of the two discrete-time models[2] for $T_s = 1.5$:

$$m_1 : \quad \mathbf{x}_{k+1} = \begin{bmatrix} 0.04 & -0.59 \\ 0.59 & 0.04 \end{bmatrix} \mathbf{x}_k + \begin{bmatrix} 0.16 \\ 0.13 \end{bmatrix} \mathbf{u}_k \tag{7}$$

$$m_2 : \quad \mathbf{x}_{k+1} = \begin{bmatrix} 0.19 & 2.64 \\ -2.64 & 0.19 \end{bmatrix} \mathbf{x}_k + \begin{bmatrix} 0.30 \\ -0.36 \end{bmatrix} \mathbf{u}_k \,. \tag{8}$$

Sampling does not change the matrices of the output equation, thus:

$$\mathbf{y}_k = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{x}_k \,. \tag{9}$$

Figure 1 visualizes the different dynamic behaviors of the hybrid system as vector fields and indicates a particular trajectory for the initial state $\mathbf{x}_0 = [-0.4, 0.4]^T$ and $\mathbf{u}_k = 0$. The dots along the trajectory indicate the sampled states $\mathbf{x}_k$.

Normally, one would use individual hybrid automata to model system components of a complex physical artifact. Their composition as a concurrent hybrid automaton will then represent the overall model. However, for clarity, we will restrict our presentation to a single component model, since this allows us to concentrate on the individual qualitative modeling assumptions and we do not loose track due to the complexity of the model itself. However, we will highlight those facts in our modeling and control scheme that are particularly relevant for the multi-component case.

Another simplifying assumption that we take at this point of our research endeavor is that we assume direct observation of the continuous state. It is our intention to relax this assumption later through the utilization of hybrid estimators similarly as it is done in standard state space control.

3 Hybrid Control and Qualitative Modeling

The task of automatic control is to actuate the hybrid system such that its hybrid state follows a particular trajectory, reaches a specific point in the state space or remains within the vicinity of a desired operational point. In order to actuate the system, hybrid control has to concurrently determine the discretely-valued actuation $\mathbf{u}_{d\,k}$ (command input) and the continuously-valued input $\mathbf{u}_k$ in order to drive the system through a continuous evolution that is interleaved with discrete mode changes. The key problem of hybrid control is, therefore, to deal with continuously valued variables ($\mathbf{x}_k$, $\mathbf{u}_k$, $\mathbf{y}_k$) and discretely valued variables ($x_{d\,k}$, $\mathbf{u}_{d\,k}$) simultaneously.

Let us demonstrate hybrid control with the hybrid model introduced above. The control objective is to traverse the system from the initial continuous state $\mathbf{x}_0 = [-0.875,\ 0.875]^T$ to a goal state that lies within a disk of radius 0.125 around the operational point $[-0.625,\ -0.625]^T$. What makes the problem complicated are the limited continuous actuation (5) and the constraints on the allowed region of the state-space (6).

Taking the qualitative view of the system's dynamics that is shown in Figure 1 we observe that the system at mode m_1 exhibits trajectories that follow counterclockwise spirals that directly traverse states from the upper left region of the state space to the lower left region, just what we desire. As a consequence, an initial guess would be to keep the system at the mode m_1 for all times, i.e. $x_{d\,k} = m_1$.

However, due to the limitation of the continuous input, we cannot find values for $\mathbf{u}_k$ between the limits 0 and 1 such that we can reach the goal region. The trajectory that achieves minimal distance for any of the states $\{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$ to the center of the goal region passes by the goal region after 2 time steps, as shown in Figure 2. Thus, we have to back-track and consider an alternative mode sequence that combines both modes of the hybrid system in a clever way.

The continuous dynamics of the model as shown in Figure 1 can be classified qualitatively in terms of a stable counterclockwise spiral for m_1 and an unstable clockwise spiral for m_2. This gives rise to the following idea: we traverse with mode m_1 until we pass by the goal region and then switch to the mode m_2 to reverse the direction and proceed towards the goal region. Because of the shape of the trajectories we guess that a trajectory segment at mode m_1 for two time

[2] Let $\mathbf{\Phi}(t)$ denote the solution of $d\mathbf{x}/dt = \mathbf{A}\mathbf{x}$. Then, one can obtain the dynamic and input matrix of the corresponding discrete-time model by $\mathbf{\Phi}(T_s)$ and $\int_0^{T_s} \mathbf{\Phi}(\tau)\mathbf{B}d\tau$, respectively.

steps, followed by a trajectory segment at mode m_2 might be a suitable mode sequence and determine values for the inputs $\mathbf{u}_k$, $k = 0, \ldots, 2$ such that the constraints on the input and the state are satisfied. This numerical optimization, that attempts to bring the state $\mathbf{x}_k, k = 1, \ldots, 3$ as close to the center of the goal region as possible, leads to the following values for the continuous input

$$\mathbf{u}_0 = 0.8, \quad \mathbf{u}_1 = 0, \quad \mathbf{u}_2 = 0.12 \,. \tag{10}$$

Figure 3 shows the trajectory of the continuous state for this mode-sequence and input actuation.

3.1 Hybrid Control through Qualitative Pre-Selection

We showed how abstracting the continuous dynamics qualitatively can help to solve the hybrid control problem. Therefore, we capture the continuous dynamics of the hybrid system in terms of a qualitative model and obtain a discretely-valued description for both, the continuous and discrete evolution of the system. Then, one can capture the control objective, that is, the desired continuous trajectory or goal state and the initial state of the hybrid system qualitatively and apply discrete search methods to determine a suitable hybrid trajectory that satisfies the control objective. This operation provides the desired mode sequence $\{x_{d\,0}, \ldots, x_{d\,N}\}$ (along with the sequence of command inputs $\{\mathbf{u}_{d\,0}, \ldots, \mathbf{u}_{d\,N-1}\}$) and a qualitatively abstracted continuous trajectory $\{\mathcal{X}_0, \ldots, \mathcal{X}_N\}$ and input sequence $\{\mathcal{U}_0, \ldots, \mathcal{U}_{N-1}\}$. This continuous part of the hybrid control problem has to be specified in more detail in order to provide the exact values of the inputs $\mathbf{u}_k$ that ought to be actuated. Because of the qualitative pre-selection, however, we obtain a hybrid control problem for a system with a fixed mode-sequence $\{x_{d\,0}, \ldots, x_{d\,N}\}$. This is equivalent to interpreting the hybrid model as a *continuously valued time-variant system* and standard methods from automatic control theory, such as Optimal Control [Bertsekas, 1995] or Model Predictive Control [Maciejowski, 2002] can be used to determine the values for the continuous inputs on a numerical basis.

This approach, however, has one major drawback. The qualitative abstraction of the hybrid model leads to a qualitative model that is subject to spurious behaviors [Kuipers, 1994] and the consecutive numerical control law deduction may be unsuccessful or may lead to unsatisfying results in terms of the control objective. In both cases, one has to backtrack to the qualitative hybrid control problem and determine an alternative result.

The described procedure for hybrid control that uses a qualitatively abstracted hybrid model to guide the search for a suitable hybrid control strategy is summarized in Figure 4.

4 Qualitative Model

The first step towards a qualitative description of the continuous dynamics of the hybrid system, that is somehow similar to a non-deterministic automaton [Lunze, 1992; 1994], is to divide the continuous domains of the variables $\mathbf{x}$, $\mathbf{u}$ and $\mathbf{y}$ into qualitatively distinct regions so that we obtain a finite

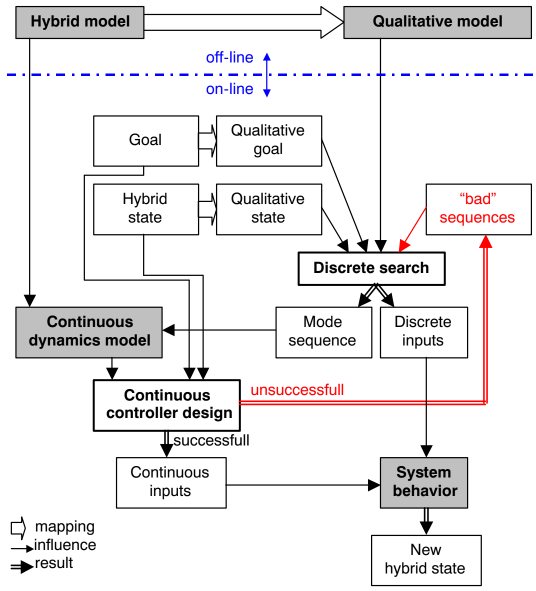


Figure 4: Hybrid control supported by a qualitative model

description of the variable's valuation. In order to keep the model compact, we select a reasonable coarse separation of qualitative values. On the one hand, we ensure that we capture the system-imposed qualitative values but also increase the granularity in order to obtain good behavior for hybrid control.

Our modeling framework is meant to work with hybrid automata of higher complexity than the example that was introduced above. First, we will compose complex systems in terms of a concurrent composition of simpler automata that model individual system components [Hofbaur and Williams, 2004]. Secondly, we will allow autonomous mode transitions, and not just commanded ones. The second fact, in particular, has implications on the state space separation since we have to introduce all qualitatively important values/borders into the qualitative domain as well.

Before we get more specific on multi-component hybrid systems, we demonstrate the approach of forming the qualitative model for our example (7-9) informally. We choose to describe the continuous state space in terms of quadratic boxes of size 0.25×0.25, as indicated by the dotted lines in the Figures 1-3. The output $\mathbf{y}_k$ provides direct observations of the state $\mathbf{x}_k$ (9) so it is reasonable to express the qualitative model in terms of state and input only. For the input $\mathbf{u}_k$ we choose two qualitative values that represent the intervals $[0, 0.5]$ and $(0.5, 1]$. In terms of notation, we will use the calligraphic letters $\mathcal{X}$ and $\mathcal{U}$ to denote the qualitative counterparts of $\mathbf{x}$ and $\mathbf{u}$, respectively. ξ_j and υ_j denote the qualitative valuation of the qualitative variable $\mathcal{X}$ and $\mathcal{U}$.

As an example, let us provide the qualitative encoding of the trajectories for the mode m_1 that start at time t_k in the upper left corner of the state space

$$-1 \leq x_{1,k} \leq -0.75, \quad 0.75 \leq x_{2,k} \leq 1 \tag{11}$$

which represents the qualitative value

$$\mathcal{X}_k = \xi_1 \tag{12}$$

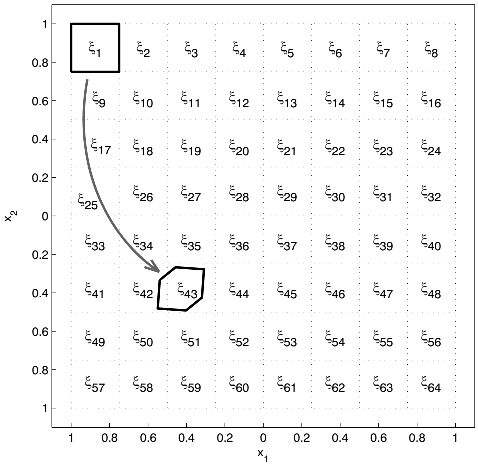


Figure 5: Reachable state space at $k + 1$ for trajectories starting from $\mathcal{X}_k = \xi_1$

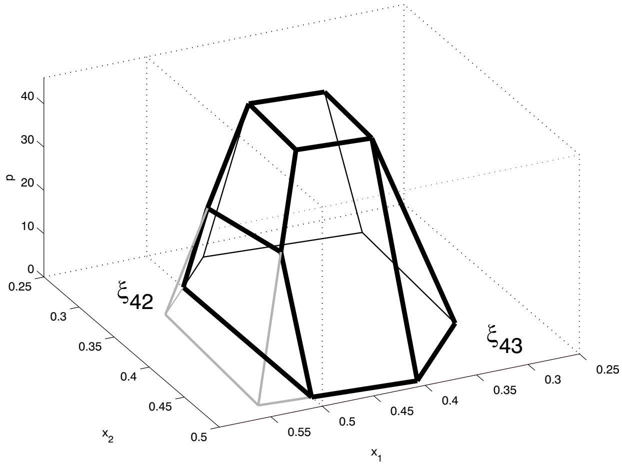


Figure 6: Probability-distribution for $\mathcal{X}_k = \xi_1 \rightarrow \mathcal{X}_{k+1} = \xi_{43}$

and an input $\mathbf{u}_k$ within the interval $(0.5,\ 1]$, qualitatively expressed as

$$\mathcal{U}_k = \upsilon_2\ . \tag{13}$$

With this specification it is straight forward to calculate the possible state at $k + 1$ using the difference equation (7). For $\mathbf{x}_{k+1}$ we obtain a region in the state space as illustrated in Figure 5 where the grid represents the qualitative quantification of the state space into $8 \times 8 = 64$ boxes $\xi_1, \ldots, \xi_{64}$.

With respect to this figure and its underlying deduction, we record two qualitative relationships as

$$\begin{array}{cccc} \text{mode} & \mathcal{X}_k & \mathcal{U}_k & \mathcal{X}_{k+1} \\ m_1 & \xi_1 & \upsilon_2 & \xi_{42} \\ m_1 & \xi_1 & \upsilon_2 & \xi_{43} \end{array}\ . \tag{14}$$

Each line of the table stands for a logical expression such as

$$(x_{d\,k} = m_1) \wedge (\mathcal{X}_k = \xi_1) \wedge (\mathcal{U}_k = \upsilon_2) \wedge (\mathcal{X}_{k+1} = \xi_{42}) \tag{15}$$

for the first line.

We observe that these trajectories are ambiguous, since for one given origin and actuation (m_1, ξ_1, υ_2) we have two different qualitative goal states (ξ_{42}, ξ_{43}). However, we additionally observe, that most of the region for $\mathbf{x}_{k+1}$ covers the qualitative value ξ_{43}. We account for this by associating each relation with a *likelihood* value L that takes this fact into account.

One way to express this likelihood is to assume uniform probability density for all valuations that correspond to the qualitative values of the variables $\mathbf{x}_k$ and $\mathbf{u}_k$ and zero probability density elsewhere. We then apply the difference equation for the mode m_i under consideration

$$\mathbf{x}_{k+1} = \mathbf{A}_i\, \mathbf{x}_k + \mathbf{B}_i\, \mathbf{u}_k$$

and calculate the resulting probability distribution through convolution [Papoulis, 1984]. For example with a 2-dimensional state space $\mathbf{x} = [x_1, x_2]^T$ and distributions $p_1(x_1, x_2)$ and $p_2(x_1, x_2)$ for $\mathbf{A}_i\, \mathbf{x}_k$ and $\mathbf{B}_i\, \mathbf{u}_k$, respectively, this is done by

$$p(x_1, x_2) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} p_1(\alpha_1, \alpha_2) \cdot p_2(x_1 - \alpha_1, x_2 - \alpha_2)\ d\alpha_2 d\alpha_1\ .$$

Figure 6 shows this operation graphically for the example relations given above (14). The resulting probability density allows us to calculate the likelihood of each relation. We obtain the values by taking the integral over all qualitative valuations of $\mathbf{x}_{k+1}$. For our example and the relations (14) we obtain likelihood values

$$\begin{array}{ccccc} \text{mode} & \mathcal{X}_k & \mathcal{U}_k & \mathcal{X}_{k+1} & L \\ m_1 & \xi_1 & \upsilon_2 & \xi_{42} & 0.08 \\ m_1 & \xi_1 & \upsilon_2 & \xi_{43} & 0.92\ . \end{array} \tag{16}$$

These likelihood values will be used as quality measure when deducing a qualitative hybrid control strategy. Of course, in terms of probabilities, they are only correct under the assumption that we have a uniform distribution at time-step k and we only perform a one-step-ahead prediction as the distribution for $\mathbf{x}_{k+1}$ is not uniform already. However, we intend to use the qualitative relations recursively to deduce N-step qualitative trajectories so using these values consecutively is, strictly speaking, an abuse of the probability notion. Nevertheless, the likelihood values provide a good indicator that tells us whether an ambiguous relation is likely to hold or whether it is just true for some very special valuations of the system's variables. Because of these relaxed requirements it is easily possible to consider even simpler measures, such as the area based measures used in [Benazero, 2003] for the estimation and reconfiguration of hybrid systems. However, we prefer the likelihood values due to their intuitive probability-like interpretation.

Up to now, we only described the qualitative abstraction of the continuous dynamics of the hybrid model. We have to include also the discrete mode transitions that can depend on the values of the input and the state variables. Without going into detail here (we did not even detail the underlying hybrid model in its full extent) we only want to note that these model properties can be formulated as relations as well so that we end up with a set of relations that directly describe the discrete dynamics of the hybrid automaton together with the qualitative abstraction of its continuous dynamics.

Of course, the number of relations increases exponentially with the number of input, output and state variables. As a compromise between model size and on-line reasoning time, we compile the possible behaviors component-wise only. Therefore, it seems reasonable that the number of variables and their qualitative state-space stays within a manageable size and on-line reasoning is responsible for deducing system-wide interactions among the system's components and to deduce qualitative trajectories that go beyond a one-step-ahead prediction.

4.1 Compilation of the Qualitative Model

Our qualitative model encodes the possible trajectories for component automata via a set of relations. Each trajectory is supplemented by a likelihood value. Apart from the observation that the assumption of uniform probability distribution at each time step k is generally not satisfied, the qualitative model for each component could be captured in terms of a non-deterministic automaton. However, we do not follow this automaton approach, since such a model would still be quite large. Borrowing from model checking [Clarke *et al.*, 1999] that deals with a very similar task, we utilize a variant of an *Ordered Binary Decision Diagram* (OBDD) [Bryant, 1986; Andersen, 1997]. This diagram, on the one hand, allows the compact representation of large sets of relations. On the other hand, the representation of the qualitative model as a graph allows us to utilize efficient search algorithms [Bertsekas, 1995].

OBDDs represent boolean expressions as a directed acyclic graph (DAG). They are often substantially more compact than traditional boolean normal forms and they can be manipulated very efficiently. For example, the following truth table for the binary variables V_1 and V_2

V_1	V_2	truth value
0	0	`true`
0	1	`true`
1	0	`false`
1	1	`true`

(17)

can be encoded into an OBDD as shown in Figure 7. This

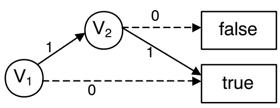


Figure 7: OBDD for truth table (17)

encoding is the result of a variable ordering $V_1 \succ V_2$ and achieves its compactness through the fact that parts of the graph are reused to represent multiple relations and that redundant parts of relations are eliminated. For example, once V_1 is set to 0 the truth value is `true`, regardless of the valuation of V_2.

Compactness of an OBDD depends on the particular ordering among the variables, where it seems a good heuristic to group together 'dependent' variables [Clarke *et al.*, 1999]. However, even more important to us is to represent structural properties of the underlying hybrid model in the ordering of variables, for this enables efficient search. This structural properties can be obtained through the causal analysis [Nayak, 1995; Travé-Massuyès and Pons, 1997] of the underlying equations. As motivation for this consider two trajectories with same $\mathbf{x}_k$ but different 'past' $(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \ldots)$, where we have depicted causal dependencies among variables in Figure 8.

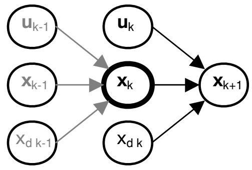


Figure 8: Causal dependencies among variables

In this figure we observe, that the 'future' of a trajectory only depends on variables at time step k and later, whereas its past is no longer relevant, since as far as further evolution is concerned it is subsumed in the value of $\mathbf{x}_k$. So, since both trajectories share common $\mathbf{x}_k$, it is reasonable to continue investigating the better one. This might seem paradoxical since it is known that qualitative abstractions of continuous systems do not possess the Markov property [Lunze, 1998]. A pure qualitative analysis of a continuous system can predict wrong (spurious) trajectories. Those trajectories, however, will be caught by the consecutive numerical optimization that fails to predict the associated continuous actuation for the system.

So, if we investigate variables in a causality-induced ordering that takes into account, for example, that the future of a particular qualitative trajectory only depends on a subset of variables while others become irrelevant, we are enabled to concentrate on 'good' trajectories, while others are dropped from further investigation. This becomes even more important, when multi-component models are taken into account, since these properties do not only apply on a temporal scale, but also on topological properties of the composition of the components.

Our qualitative model records only valid relations, i.e. ones that evaluate to `true` and which are associated with a particular likelihood value. Furthermore, our qualitative variables can usually take on more than just 2 binary values so that we have to utilize a modified OBDD concept. In order to deal with non-binary symbolic variables, we take a standard approach and replace each symbolic variable $\mathcal{V}$ with a domain size $|\text{dom}(\mathcal{V})| > 2$ by m binary variables that encode the $|\text{dom}(\mathcal{V})| \leq 2^m$ different valuations. The modifications of the OBDD to incorporate likelihood values is more involving and outlined in the remainder of this section.

First of all, we intend to replace the binary truth values of the terminal nodes of the OBDD by likelihood values. Compactness of OBDDs result from the fact that several relations link to common terminal nodes with shared truth values. Now, introducing terminal nodes for all likelihood values of all relations is surely not a good idea. Therefore, we approximate the likelihood values in that we group them into few distinct ranges only.

Assume, for example, the following variant of the truth table (17), where we use likelihood values for the `true`-

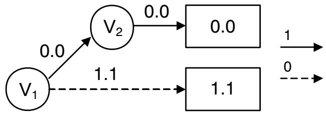


Figure 9: Labeled directed acyclic graph for (20)

relations:

$$\begin{array}{ccc} V_1 & V_2 & L \\ 0 & 0 & 0.27 \\ 0 & 1 & 0.41 \\ 1 & 1 & 0.98 \end{array} \quad (18)$$

In order to keep the OBDD-like directed acyclic graph compact, we group the three likelihood values into two groups and separate the range $[0,\ 1]$ at 0.5. Of course, the separation of the probability space into n equally sized intervals is the simplest method one can think of. It is also possible to analyze the distribution of likelihood values and separate them accordingly. We then replace the likelihood value L by the center of the interval it belongs to, thus we obtain $\bar{L}_1 = \bar{L}_2 = 0.25$ for the first two relations of (18), and $\bar{L}_3 = 0.75$ for the third one. Of course, this leads to a coarse quantification of the likelihood values that provides the distinction between transitions that differ in the magnitude of their likelihood values, exactly what we are looking for for our qualitative mode-sequence pre-selection mechanism.

Our overall goal is to have a compact graph that represents all possible trajectories in a weighted sense. We intend to use standard search algorithms, such as shortest path, to find the 'best' trajectories. As a preparation for this fact, we calculate the negative logarithm of the unified likelihood values $\bar{c}_i = -\ln(\bar{L}_i)$ and obtain the cost values as

$$c_i := \bar{c}_i - \min_j \bar{c}_j \quad (19)$$

which normalizes the best possible cost value to zero. This leads to the table

$$\begin{array}{ccc} V_1 & V_2 & c \\ 0 & 0 & 1.1 \\ 0 & 1 & 1.1 \\ 1 & 1 & 0.0 \end{array} \quad (20)$$

This table enables us to use a standard OBDD generation procedure to obtain the directed acyclic graph (DAG) that encodes the valid qualitative trajectories of the hybrid model. To integrate cost values into that graph, we start with labeling the terminal nodes with the corresponding cost-values c. Afterwards, each node in the graph recursively is assigned the minimum cost value of its children, so that each node is labeled with the lowest cost-value of all terminal nodes reachable from it. With this we bring the information whether particular variable-assignments are more likely than others closer to the root of the graph. Finally, all edges are labeled with the difference between cost values assigned to their adjacent nodes so that all path costs in the graph from its root to a leaf represent the cost value of the corresponding variable-assignment. The resulting *trajectory-DAG* of (20) is shown in Figure 9.

5 Example

We provide a short summary of compiling the qualitative model for the example (7-9) given in Section 2. The hybrid model contains three continuously-valued variables ($\mathbf{x}$, $\mathbf{u}$, $\mathbf{y}$) and two discretely-valued ones (x_d, $\mathbf{u}_d$). Output $\mathbf{y}$ and state $\mathbf{x}$ are identical for all operational modes. Similarly, there is an identity between a particular value of the command input and the corresponding operational mode. Thus, modeling the discrete evolution among operational modes becomes trivial and is omitted. Thus, we only have to build a qualitative model for equations (7-8). To deduce the qualitative model from these equations, we first encode the domains of the continuously valued variables ($\mathbf{x}$, $\mathbf{u}$) into distinct qualitative values. To remain conform with the presentation in previous sections, we describe the state space $\mathbf{x}$ (that is, $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$) in terms of 64 quadratic boxes of size 0.25×0.25 (Figure 5) and the input $\mathbf{u}$ (that is, $\mathbf{u}_k$) in terms of the two intervals $[0, 0.5]$ and $(0.5, 1]$.

Next, we record all trajectories for time $t_k \rightarrow t_{k+1}$ that are allowed by (Figures 1-3) in terms of their qualitative values and obtain

$$785 \text{ relations}$$

among variables $\mathcal{U}_k$, $x_{d\,k}$, $\mathcal{X}_k$ and $\mathcal{X}_{k+1}$.

To compile these relations into the trajectory graph, we have to determine an ordering among the qualitative abstractions of the variables $\mathbf{x}_k, \mathbf{x}_{k+1}, x_{d\,k}, \mathbf{u}_k$. For one time step we obtain the variable ordering based on the causal analysis shown in Figure 8 as[3]:

$$\mathbf{x}_k \succ x_{d\,k} \succ \mathbf{u}_k \succ \mathbf{x}_{k+1} \ .$$

Furthermore, we group the likelihood values into 4 discrete ranges $[0, 0.25)$, $[0.25, 0.5)$, $[0.5, 0.75)$ and $[0.75, 1]$ (with corresponding cost values, for example $-\ln(0.125)$ for the range $[0, 0.25)$) and introduce 6 binary variables to represent the 64 qualitative values for the state $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$, respectively. The resulting DAG is then used recursively to reason about a finite prediction horizon of $N > 1$ steps.

In our example, we want to use a horizon of 3 steps from t_0 to t_3 and obtain a compiled trajectory graph with

$$782 \text{ nodes}$$
$$1161 \text{ edges} \ .$$

As above, the control goal is to bring the state $\mathbf{x}_0 = [-0.875, 0.875]$ to $\mathbf{x}_3$, which is inside a disk of radius 0.125 around the continuous state $[-0.625, -0.625]$, whilst satisfying $0 \leq \mathbf{u}_k \leq 1$ and $-1 \leq x_{i,k} \leq 1$.

Search for the best sequence of operational modes $x_{d\,1}$, $x_{d\,2}$, $x_{d\,3}$, that meets all of the control constraints leads to

$$x_{d\,1} = m_1, \;\; x_{d\,2} = m_1, \;\; x_{d\,3} = m_2 \ .$$

With this mode-sequence, subsequent numerical optimization of the actuation $\mathbf{u}_0$, $\mathbf{u}_1$, $\mathbf{u}_2$ provides the trajectory already depicted in Figure 3.

[3] The ordering of scalars among vectors is legitimate since our qualitative framework encodes a vector in terms of a single qualitative variable.

6 Conclusion

For simple hybrid control problems, deducing a 'good' sequence of operational modes by reasoning about possible trajectories can be done intuitively. This paper outlines an automated reasoning scheme that makes qualitative pre-selection of feasible mode sequences for hybrid control applicable to more complex systems.

Basis for this approach is a qualitative model that facilitates efficient search and compact storage. We obtain this through the formulation of an approximate likelihood value for trajectory segments and an off-line compilation scheme that provides an OBDD like encoding of the model. This model then facilitates the efficient on-line deduction of the hybrid control law and is intended to be integrated into a more general hybrid automation framework in the future.

Acknowledgments

The authors thank the anonymous reviewers for their valuable comments and suggestions.

References

[Andersen, 1997] H.R. Andersen. *An Introduction to Binary Decision Diagrams*. Lecture notes for 49285 Advanced Algorithms E97, Department of Information Technology, Technical University of Denmark, 1997.

[Askari *et al.*, 1999] J. Askari, B. Heiming, and J. Lunze. Controller reconfiguration based on a qualitative model: A solution of three-tanks benchmark problem. In *Proceedings of the European Control Conference, Karlsruhe, Germany*, 1999.

[Bemporad and Morari, 1999] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics and constraints. *Automatica*, 35(3):407–427, 1999.

[Benazero, 2003] E. Benazero. *Diagnosis and reconfiguration based on hybrid concurrent models - Application to autonomous satellites*. PhD thesis, Rapport LAAS No03386. Universit Paul Sabatier, Toulouse, 2003.

[Bertsekas, 1995] D. Bertsekas. *Dynamic Programming and Optimal Control*, volume 2. Athena Scientific, 1995.

[Bryant, 1986] R. Bryant. Graph-based algorithms for boolean function manipulation. *IEEE Transactions on Computers*, C-35(8):677–691, 1986.

[Clarke *et al.*, 1999] E.M. Clarke, O. Grumberg, and D.A. Peled. *Model Checking*. MIT Press, 1999.

[Hamscher *et al.*, 1992] W. Hamscher, L. Console, and J. DeKleer, editors. *Readings in Model-Based Diagnosis*. Morgan Kaufmann, 1992.

[Hofbaur and Williams, 2004] M. W. Hofbaur and B. C. Williams. Hybrid estimation of complex systems. *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, 34(5):2178–2191, October 2004.

[Hofbaur, 2005] M. W. Hofbaur. *Hybrid Estimation of Complex Systems*, volume 319 of *Lecture Notes in Control and Information Sciences (LNCIS)*. Springer Verlag, 2005.

[Kuipers, 1994] B.J. Kuipers. *Qualitative Reasoning: Modeling and Simulation with incomplete Knowledge*. MIT Press, Cambridge, MA, 1994.

[Lunze, 1992] J. Lunze. Qualitative modeling of continuous variable systems by means of non-deterministic automata. *Journal of Intelligent System Engineering*, 1:22–30, 1992.

[Lunze, 1994] J. Lunze. Qualitative modeling of linear dynamical systems with quantized state measurements. *Automatica*, 30(3):417–431, 1994.

[Lunze, 1998] J. Lunze. On the markov property of quantised state measurement sequences. *Automatica*, 34(11):1439–1444, November 1998.

[Maciejowski, 2002] J. M. Maciejowski. *Predictive Control with Constraints*. Pearson Education Ldt., Essex, UK, 2002.

[Nayak, 1995] P. Nayak. *Automated Modelling of Physical Systems*. Lecture Notes in Artificial Intelligence. Springer, 1995.

[Papoulis, 1984] A. Papoulis. *Probability, Random Variables and Stochastic Processes*. Mc Graw Hill, 2 edition, 1984.

[Travé-Massuyès and Pons, 1997] L. Travé-Massuyès and R. Pons. Causal ordering for multiple mode systems. In *Proceedings of the 11th workshop on Qualitative Reasoning (QR97)*, pages 203–214, 1997.