Optimal stabilization of hybrid systems using a set oriented approach

Lars Grüne
Mathematisches Institut
Universität Bayreuth
95440 Bayreuth, Germany
lars.gruene@uni-bayreuth.de

Oliver Junge
Center for Mathematical Sciences
Munich University of Technology
85747 Garching, Germany
junge@ma.tum.de

***Abstract*— We demonstrate how a recently developed new numerical technique for the construction of approximately optimal stabilizing feedback laws can naturally be extended in order to handle nonlinear hybrid systems with discrete control inputs. The idea of the method is to explicitly construct a finite graph model of the original hybrid system and to use shortest path algorithms in order to compute the optimal value function and the associated feedback law. As a numerical example, we reconsider the construction of a switched DC/DC power converter from [12].**

I. Introduction

The control of hybrid systems is a topic which received considerable interest during the last years. The mixture of continuous and discrete components in such systems creates severe difficulties in their analytical and numerical treatment, for instance when stabilizability or reachability problems are to be solved or when optimal control methods are to be applied. Particular progress has been made in the field of piecewise or switched linear systems, where different kinds of optimization techniques turned out to be applicable, see, e.g., the monographs [1] or [9] and the references therein.

In this paper we focus on the optimal stabilization of a general class of nonlinear discrete time hybrid systems, possessing discrete and continuous state variables and a discrete control value set. We consider an approach for discrete time and continuous state systems presented recently in [10], [7] and show how to modify the method in order to cover the hybrid setup. The method consists of a set oriented discretization of the state space and a subsequent representation of the system as a finite directed graph. On this discretized level, Dijkstra's shortest path algorithm can then be applied in order to solve the problem.

A technique that is closely related to our approach is described in [2] for time continuous systems using a so-called bisimulation in order to construct a finite automaton. They rely on the existence (and knowledge) of a particular number of first integrals that are needed to construct the bisimulation partition. This approach has been pursued in the context of hybrid systems in, e.g., [11].

The organization of this paper is as follows: in Section II we formulate our problem and in Section III we present our computational approach for the optimal value function of the problem. Based on this, in Section IV we show how to construct the optimal stabilizing feedback and in Section V we discuss a local error estimation technique for our discretization. In Section VI we present an application of our method to a switched DC/DC power converter model in order to illustrate the performance of the resulting feedback. Finally, Section VII presents conclusions and in particular a comparison to the related approach from [12], [14].

II. Problem formulation

We consider the problem of optimally stabilizing the continuous state component x of a discrete-time nonlinear hybrid control system given by

$$\begin{array}{rcl} x_{k+1} & = & f_c(x_k, y_k, u_k) \\ y_{k+1} & = & f_d(x_k, y_k, u_k) \end{array} \qquad k = 0, 1, \ldots, \qquad (1)$$

with continuous state dynamics $f_c : X \times Y \times U \to X \subset \mathbb{R}^n$ and discrete state dynamics $f_d : X \times Y \times U \to Y$. Here the set U of possible control inputs is finite[1], the set $X \subset \mathbb{R}^n$ of continuous states is compact and the set Y of discrete states (or modes) is an arbitrary finite set. The solutions of (1) for initial values $x_0 = x$, $y_0 = y$ and control sequence $\mathbf{u} = (u_0, u_1, \ldots) \in U^{\mathbb{N}}$ are denoted by $x_k(x, y, \mathbf{u})$ and $y_k(x, y, \mathbf{u})$, respectively, and we assume that for each $k \geq 0$ the map $x_k(\cdot, y, \mathbf{u})$ is continuous for each $y \in Y$ and each $\mathbf{u} \in U^{\mathbb{N}}$. Note that if f_d does not depend on x, then this is equivalent to $f_c(\cdot, y, u) : X \times Y \to \mathbb{R}^d$ being continuous for each $y \in Y$, $u \in U$.

The class (1) of hybrid models is quite general. For instance, it includes models without discrete state space component y when $f_c(x, y, u) = f_c(x, u)$ by setting $Y = \{0\}$ and $f_d \equiv 0$, in which case the only "hybrid" structure is given by the discrete nature of the finite control value set U. Another specialization of (1) appears if $f_c(x, y, u) = f_c(x, y)$ and $f_d(x, y, u) = f_d(y, u)$ in which case the continuous state plant is controlled solely by the discrete variable y which in turn is determined by the discrete dynamics f_d, which may be realized, e.g., by a discrete automaton. Finally, for general f_c and $f_d(x, y, u) = f_d(x)$ we obtain a hybrid system with state dependent switching.

Given a *target set* $T \subset X$, the goal of the optimization problem we want to solve is to find a control sequence

[1] If desired, continuous control values could also be included and treated with the discretization technique described in [10], [7].

$u_k, k = 0, 1, 2, \ldots$, such that $x_k \to T$ as $k \to \infty$, while minimizing the accumulated continuous instantaneous cost $g : X \times Y \times U \to [0, \infty)$ with $g(x, y, u) > 0$ for all $x \notin T$, all $y \in Y$ and all $u \in U$.

We assume that (1) is locally asymptotically controllable to T, i.e., there exists a $\mathcal{KL}$–function[2] β and a neighborhood $\mathcal{N}(T) \subset X$ of T, such that for each $x \in \mathcal{N}(T)$ there exists a control sequence $\mathbf{u} \in U^{\mathbb{N}}$ with

$$d(x_k(x, y, \mathbf{u}), T) \leq \beta(\|x\|, k) \quad \text{for all } y \in Y.$$

By $\mathcal{U}(x, y) = \{\mathbf{u} \in U^{\mathbb{N}} : x_k(x, y, \mathbf{u}) \to T\}$ we denote the set of asymptotically controlling sequences for $(x, y) \in X \times Y$ and by $S = \{(x, y) \in X \times Y : \mathcal{U}(x, y) \neq \emptyset\}$ the *stabilizable subset*. The *accumulated cost* along a controlled trajectory is given by

$$J(x, y, \mathbf{u}) = \sum_{k=0}^{\infty} g(x_k(x, y, \mathbf{u}), y_k(x, y, \mathbf{u}), u_k) \in [0, \infty]$$

and we assume that g is chosen such that this sum is finite for each $(x, y) \in S$ and each $\mathbf{u} \in \mathcal{U}(x, y)$ for which

$$d(x_{k_0+k}(x, y, \mathbf{u}), T) \leq \beta(\|x_{k_0}(x, y, \mathbf{u})\|, k)$$

holds for some $k_0 \in \mathbb{N}$ and all $k \in \mathbb{N}$ (suitable conditions on g can be formulated in terms of β, see [8] or [5, Section 7.2] for details).

Our goal is to construct an approximate *optimal feedback* $u : S \to U$ such a suitable approximate asymptotic stability property for the resulting closed loop system holds. The construction will be based on an approximation of the *(optimal) value function* $V : S \to [0, \infty]$,

$$V(x, y) = \inf_{\mathbf{u} \in \mathcal{U}(\mathbf{x}, \mathbf{y})} J(x, y, \mathbf{u})$$

which will act as a Lyapunov function. For an appropriate choice of g this function is continuous in x at least in a neighborhood of T [8].

In order to simplify the notation we write $z = (x, y)$, $Z = X \times Y$ and denote the dynamics (1) briefly by

$$z_{k+1} = f(z_k, u_k). \tag{2}$$

III. Computational approach

In this section we discuss a set oriented numerical method for the computation of V which was developed in [10]. The method relies on the observation that one may formulate the above discrete-time optimal control problem equivalently as the problem of finding a *shortest path* within a directed weighted graph: Consider the graph $G = (Z, E)$, where the set E of edges of G is given by

$$E = \{(z_1, z_2) \in Z \times Z \mid \exists u \in U : z_2 = f(z_1, u)\},$$

and for every edge $e = (z, f(z, u)) \in E$ the weight $w(e) \in [0, \infty)$ is given by $w(e) = \min_{u \in U} g(z, u)$. A *path* in G is a sequence $p = (e_k)_{k \in \mathbb{N}} \in E^{\mathbb{N}}$ of edges of G. The *length* $w(p)$ of a path $p = (e_k)_{k \in \mathbb{N}}$ is defined to be

$$w(p) = \sum_{k=0}^{\infty} w(e_k) \in [0, \infty].$$

Every path in G uniquely corresponds to a (controlled) trajectory of (2). By construction, for a given path, the cost $J(z, \mathbf{u})$ of the associated trajectory equals the length of this path. Thus, when asking for the infimum of $J(z_1, \mathbf{u})$ over all control sequences $\mathbf{u}$, we can equivalently ask for the infimum of $w(p)$ over all paths p in G that start in z_1 (i.e. such that if $p = (e_k)_{k \in \mathbb{N}}$, e_0 is of the form $e_0 = (z_1, z_2)$ for some $z_2 \in Z$).

We are now going to construct a finite graph $G_{\mathcal{P}} = (\mathcal{P}, E_{\mathcal{P}})$ — which should be viewed as an approximation to the graph G — in order to compute an approximation to V. The idea is that on $G_{\mathcal{P}}$ we can apply standard algorithms for computing paths of shortest length. A typical algorithm of this type is *Dijkstra's algorithm* [4].

The finite approximation to G is constructed as follows: Let $\mathcal{Q}$ be a *partition* of the continuous state set X, that is a finite collection of compact subsets $Q_i \subset X$, $i = 1, \ldots, r$, with $\cup_{i=1}^{r} Q_i = X$, and $m(Q_i \cap Q_j) = 0$ for $i \neq j$ (where m denotes Lebesgue measure). Then the sets

$$\mathcal{P} := \{Q_i \times \{y\} \mid Q_i \in \mathcal{Q}, y \in Y\} \tag{3}$$

form a partition of the product state space $Z = X \times Y$.

Define the graph

$$G_{\mathcal{P}} = (\mathcal{P}, E_{\mathcal{P}}), \tag{4}$$
$$E_{\mathcal{P}} = \{(P_i, P_j) \in \mathcal{P} \times \mathcal{P} \mid f(P_i, U) \cap P_j \neq \emptyset\}, \tag{5}$$

where the edge $e = (P_i, P_j)$ carries the weight

$$w(e) = \min_{z \in P_i, u \in U} \{g(z, u) \mid f(z, u) \in P_j\}. \tag{6}$$

We use $G_{\mathcal{P}}$ to find an approximation to the optimal value function V. For any $z \in Z$ there is a least one subset $P \in \mathcal{P}$ containing z. The approximation for $V(z)$ will be the length $w(p) = \sum_{k=0}^{\ell} w(e_k)$ of a *shortest path* $p = (e_0, \ldots, e_\ell)$, $e_k \in E_{\mathcal{P}}$, from a node P with $x \in P$ to a node $P' \in \mathcal{P}$ that has nonempty intersection with T, i.e. we approximate $V(x)$ by

$$V_{\mathcal{P}}(z) = \min\{w(p) \mid p \text{ is a path from a set } P, z \in P, \text{ to a set } P' \text{ with } (T \times Y) \cap P' \neq \emptyset\}.$$

Convergence

Let $(\mathcal{P}(l))_l$ be a nested sequence of partitions of Z (i.e. for every l, each element of $\mathcal{P}(l+1)$ is contained in an element of $\mathcal{P}(l)$). It is easy to see that $V_{\mathcal{P}}(z) \leq V(z)$ for any partition $\mathcal{P}$ of Z and all $z \in Z$. In fact, it can be shown that for $z \in S$, $V_{\mathcal{P}(l)}$ converges to V as $l \to \infty$; the corresponding proof for purely continuous state models in [10], [7] is easily extended to our hybrid setting.

[2] As usual, a function $\gamma : [0, \infty) \to [0, \infty)$ is of class $\mathcal{K}$ if it is continuous, zero at zero and strictly increasing. A function $\beta : [0, \infty) \times [0, \infty) \to [0, \infty)$ is of class $\mathcal{KL}$ if it is continuous, of class $\mathcal{K}$ in the first variable and strictly decreasing to 0 in the second variable.

Implementation

The computation of $V_\mathcal{P}$ breaks down into three steps:

1) Construction of a suitable partition $\mathcal{P}$;
2) Construction of $G_\mathcal{P}$;
3) Computation of $V_\mathcal{P}$ by applying Dijkstra's algorithm to $G_\mathcal{P}$.

In the numerical realization we always let X be a box in $\mathbb{R}^d$ and construct a partition $\mathcal{Q}$ of X by dividing X uniformly into smaller boxes from which $\mathcal{P}$ is then derived via (3). We realize this division by repeatedly bisecting the current division (changing the coordinate direction after each bisection). The resulting sequence of partitions can efficiently be stored as a binary tree — see [3] for details.

Once $\mathcal{P}$ has been constructed, we need to compute the set $E_\mathcal{P}$ of edges of $G_\mathcal{P}$, as well as the weight $w(e)$ for every edge $e \in E_\mathcal{P}$. Here we approximate $E_\mathcal{P}$ by

$$\tilde{E}_\mathcal{P} = \{(P_i, P_j) \mid f(\tilde{P}_i, U) \cap P_j \neq \emptyset\},$$

where $\tilde{P}_i \subset P_i$ is a *finite* set of *test points*. For example, one may choose this set as points on an equidistant grid. Correspondingly the weight $w(e)$ on $e = (P_i, P_j)$ is approximated by

$$\tilde{w}(e) = \min_{z \in \tilde{P}_i, u \in U} \{g(z,u) \mid f(z,u) \in P_j\}.$$

Again, we refer to [3] and [10] for details.

IV. Constructing the feedback

For the construction of the approximately optimal feedback law we use the classical dynamic programming technique. It follows from standard dynamic programming arguments that the exact optimal value function V satisfies

$$V(z) = \min_{u \in U}\{g(z,u) + V(f(z,u))\}$$

and that an optimal feedback law u is given by the control $u(z)$ minimizing the right hand side of this equation.

For the construction of our feedback law we will use this fact, replacing V by its approximation $V_\mathcal{P}$. Thus for each point $z \in S$ we define

$$u_\mathcal{P}(z) = \operatorname{argmin}_{u \in U}\{g(z,u) + V_\mathcal{P}(f(z,u))\} \qquad (7)$$

The following theorem shows in which sense this feedback is approximately optimal.

Theorem 1: Consider a sequence of partitions $\mathcal{P}(l)$, $l \in \mathbb{N}$ and let $D \subseteq S$ be an open set with the following properties:

(i) $T \times Y \subset D$

(iii) For each $\varepsilon > 0$ there exists $l_0(\varepsilon) > 0$ such that the inequality

$$V(z) - V_{\mathcal{P}(l)}(z) \leq \varepsilon$$

holds for all $z \in D$ and all $l \geq l_0(\varepsilon)$.

Let $c > 0$ be the largest value such that the inclusion $D_c(l) := V_{\mathcal{P}(l)}^{-1}([0,c]) \subset D$ holds for all $l \in \mathbb{N}$. (Note that $c > 0$ if $\mathcal{P}(1)$ is chosen appropriately.)

Then there exists $\varepsilon_0 > 0$ and a function $\delta : \mathbb{R} \to \mathbb{R}$ with $\lim_{\alpha \to 0} \delta(\alpha) = 0$, such that for all $\varepsilon \in (0, \varepsilon_0]$, all $l \geq l_0(\varepsilon/2)$, all $\eta \in (0,1)$ and all $z_0 \in D_c(l)$ the trajectory z_i generated by

$$z_{i+1} = f(z_i, u_{\mathcal{P}(l)}(z_i))$$

satisfies

$$V(z_i) \leq V(z_0) - (1-\eta)\sum_{j=0}^{i-1} g(z_j, u_{\mathcal{P}(l)}(z_j)),$$

for all i such that $V(z_i) \geq \delta(\varepsilon/\eta) + \varepsilon$.

V. Error estimation

>From a practical point of view, Theorem 1 does not give much information about the structure of the partition $\mathcal{P}$ which is needed in order to achieve a desired level of accuracy of the optimal value function.

Let $S_0 = \{z \in Z : V(z) < \infty\}$. For $z \in S_0$ consider the error function

$$e(z) = \min_{u \in U}\{g(z,u) + V_\mathcal{P}(f(z,u))\} - V_\mathcal{P}(z).$$

Note that by definition of $V_\mathcal{P}$ we have $e(z) \geq 0$. Furthermore,

$$e(z) \leq V(z) - V_\mathcal{P}(z), \quad z \in S_0.$$

For $D_c = V_\mathcal{P}^{-1}([0,c])$ define

$$\delta(\varepsilon) := \sup_{z \in C_\varepsilon} V(x),$$

where $C_\varepsilon := \{z \in D_c \,|\, g_0(z) \leq \varepsilon\}$ and $g_0(z) := \inf_{u \in \mathcal{U}(z)} g(z,u)$. Note that $\delta(\varepsilon) \to 0$ as $\varepsilon \to 0$ because C_ε shrinks down to 0 and V is continuous in x around T with $V(0,y) = 0$.

Theorem 2: Consider a partition $\mathcal{P}$ and a sublevel set $D_c = V_\mathcal{P}^{-1}([0,c])$ for some $c > 0$. Assume that the error estimate e satisfies

$$e(z) \leq \max\{\eta g_0(z), \varepsilon\} \qquad (8)$$

for all $z \in D_c$, some $\varepsilon > 0$ and some $\eta \in (0,1)$.

Then the trajectory z_i generated by

$$z_{i+1} = f(z_i, u_\mathcal{P}(z_i)) \qquad (9)$$

for each $x_0 \in D_c$ satisfies

$$V_\mathcal{P}(z_i) \leq V_\mathcal{P}(z_0) - (1-\eta)\sum_{j=0}^{i-1} g(z_i, u_{\mathcal{P}(l)}(z_i)), \qquad (10)$$

for all i such that $V_\mathcal{P}(z_i) \geq \delta(\varepsilon/\eta) + \varepsilon$.

If the main purpose of the kind of optimal control problems treated in our setup is the derivation of asymptotically stabilizing feedback laws one might ask to relax the strict "approximate optimality" condition by looking only for a feedback which — although far from optimal — still ensures approximate asymptotic stability in a suitable sense. In this case, it may be a good compromise to choose a relatively large $\eta \in (0,1)$. This way we slow down the convergence of the trajectories to the (neighborhood of the) origin, but in turn the problem becomes numerically easier and can be solved on a coarser partition. Such relaxations of the optimality conditions have recently been used also for other dynamic programming formulations of optimal control problems, see [12], and can considerably reduce the computational cost.

The framework from [13] allows the conclusion of asymptotic stability in our framework, as stated in the following corollary.

Corollary 1: Let the assumptions of Theorem 2 be satisfied. Then for any $\eta \in (0,1)$ the feedback law $u_{\mathcal{P}}$ renders the x_i–component of the closed loop system

$$z_{i+1} = f(z_i, u_{\mathcal{P}}(z_i))$$

with $z_i = (x_i, y_i)$ practically asymptotically stable on D_c, i.e., there exists a $\mathcal{KL}$–function β depending on g_0 and η, with the property that for any $\delta > 0$ there exists $\varepsilon > 0$ such that

$$d(x_{i+1}, T) \le \beta(\|x_0\|, t) + \delta$$

holds for each $z_0 = (x_0, y_0) \in D_c$ and all partitions for which the error estimate e satisfies the assumption of Theorem 2 with the given ε.

In general one cannot expect robustness of the feedback law even for arbitrarily small perturbations $\tilde{f}$ of f if the controller design is based on the discontinuous (Lyapunov) function $V_{\mathcal{P}}$. However, using the concept of multivalued games, it is possible to systematically account for additional (bounded) perturbations, see [6].

VI. Numerical example: A switched voltage controller

In order to demonstrate the effectiveness of our approach we reconsider an example from [12]: A switched power controller for DC to DC conversion. Within the controller, a semiconductor device is switching the polarity of a voltage source in order to keep the load voltage as constant as possible. The mathematical model is given by

$$\begin{aligned}
\dot{x}_1 &= \frac{1}{C}(x_2 - I_{load}) \\
\dot{x}_2 &= -\frac{1}{L}x_1 - \frac{R}{L}x_2 + \frac{1}{L}uV_in \qquad (11)\\
\dot{x}_3 &= V_{ref} - x_1
\end{aligned}$$

(cf. Fig. VI), where $u \in \{-1, 1\}$ is the control input. In the following numerical experiment we use the same parameter values as given in [12]. Note that this is an example of a system where no discrete variable is present (i.e, we can identify x with z) and the hybrid structure is solely represented by the switching control, i.e., by the finiteness of U.

The corresponding discrete time system is given by the time-h-map ϕ^h ($h = 0.1$ in our case) of (11), with the control input held constant during this sample period. The cost function is

$$g(x,u) = q_P(\phi_1^h(x) - V_{ref}) + q_D(\phi_2^h(x) - I_{load}) + q_I\phi_3^h(x).$$

The third component in (11) is only being used in order to penalize a large L^1-error of the output voltage. We slightly simplify the problem (over its original formulation in [12]) by using $x_3 = 0$ as initial value in each evaluation of the discrete map. Correspondingly, the map reduces to a two-dimensional one.

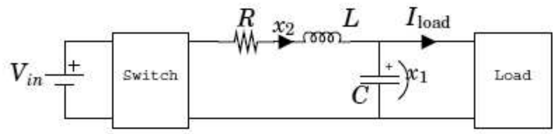

Fig. 1. A switched DC/DC converter (taken from [12]).

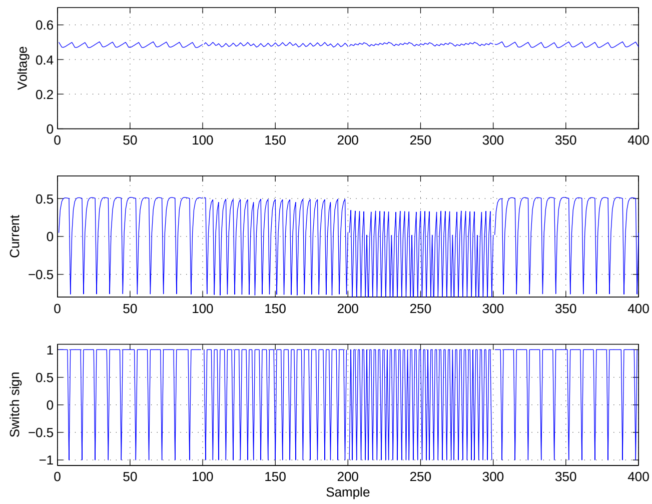

Fig. 2. Simulation of the controlled switched power converter.

Confining our domain of interest to the rectangle $X = [0,1] \times [-1,1]$, our target set is given by $T = \{V_{ref}\} \times [-1,1]$. For the construction of the finite graph, we employ a partition of X into 64×64 equally sized boxes. We use 4 test points in each box, namely their vertices, in order to construct the edges of the graph.

Using the resulting approximate optimal value function and the associated feedback, we repeated the stabilization experiment from [12], where the load current is changed after every 100 iterations. Figure VI shows the result of this simulation, proving that our controller stabilizes the system as requested.

VII. Conclusion

We have presented a graph theoretic numerical method for the optimal feedback stabilization of hybrid systems. Due to the special kind of set oriented discretization, discrete control value sets and discrete state variable components are easily included in our framework.

Our approach is complementary to the relaxed dynamic programming approach in [12], [14] in the following sense: in [12], [14] an approximation to the optimal value function via classes of highly regular functions (typically certain polynomials) is used while in our approach a highly irregular function, i.e., a piecewise constant function is used. As a consequence, the relaxed dynamic programming approach is able to treat problems with rather high dimensional state space provided the optimal value function is close to a member of this class of approximating functions (cf. [14, Example 5]) while our method is confined to rather low dimensional state space allowing, however, for nonsmooth and even discontinuous optimal value functions.

Due to the conceptional similarities, a natural next step would be a combination of our set oriented and graph theoretic approach with the relaxed dynamic programming methods from [12], [14] which captures the advantages of both approaches. This is the topic of future research.

REFERENCES

[1] F. Borrelli, *Constrained optimal control of linear and hybrid systems*, ser. Lecture Notes in Control and Information Sciences. Berlin: Springer-Verlag, 2003, vol. 290.

[2] M. Broucke, “A geometric approach to bisimulation and verification of hybrid systems,” in *HSCC 1999, LNCS*, F. V. J. van Schuppen, Ed. Springer, 1999, vol. 1569, pp. 61–75.

[3] M. Dellnitz and A. Hohmann, “A subdivision algorithm for the computation of unstable manifolds and global attractors,” *Numer. Math.*, vol. 75, no. 3, pp. 293–317, 1997.

[4] E. Dijkstra, “A note on two problems in connection with graphs,” *Numer. Math.*, vol. 5, pp. 269–271, 1959.

[5] L. Grüne, *Asymptotic Behavior of Dynamical and Control Systems under Perturbation and Discretization*, ser. Lecture Notes in Mathematics, Vol. 1783. Springer–Verlag, 2002.

[6] L. Grüne and O. Junge, “Global optimal control of perturbed systems,” Preprint, Universität Bayreuth, submitted, 2005.

[7] ——, “A set oriented approach to optimal feedback stabilization,” *Systems Control Lett.*, vol. 54, no. 2, pp. 169–180, 2005.

[8] L. Grüne and D. Nešić, “Optimization based stabilization of sampled–data nonlinear systems via their approximate discrete–time models,” *SIAM J. Control Optim.*, vol. 42, pp. 98–122, 2003.

[9] M. Johansson, *Piecewise linear control systems. A computational approach*, ser. Lecture Notes in Control and Information Sciences. Berlin: Springer-Verlag, 2003, vol. 284.

[10] O. Junge and H. M. Osinga, “A set oriented approach to global optimal control,” *ESAIM Control Optim. Calc. Var.*, vol. 10, no. 2, pp. 259–270 (electronic), 2004.

[11] G. Lafferriere, G. Pappas, and S. Sastry, “Hybrid systems with finite bisimulations,” in *Hybrid Systems V*, P. Antsaklis, W. Kohn, M. Lemmon, A. Nerode, and S. Sastry, Eds. Lecture Notes in Computer Science, Springer, 1999, vol. 1567, pp. 186–203.

[12] B. Lincoln and A. Rantzer, “Relaxing dynamic programming,” *IEEE Trans. Autom. Control*, to appear.

[13] D. Nešić and A. R. Teel, “A framework for stabilization of nonlinear sampled-data systems based on their approximate discrete-time models,” *IEEE Trans. Automat. Control*, vol. 49, no. 7, pp. 1103–1122, 2004.

[14] A. Rantzer, “On relaxed dynamic programming in switching systems,” *IEE Proceedings, Special Issue on Hybrid Systems*, to appear.