52nd IEEE Conference on Decision and Control
December 10-13, 2013. Florence, Italy

Feedback Control Law Generation for Safety Controller Synthesis

Andrew K. Winn and A. Agung Julius

***Abstract*— In recent papers we have focused on the task of safety controller synthesis, that is, designing a controller that will take the system from any point within a compact set of initial states to a point inside a set of acceptable goal states, while never entering any state that is deemed unsafe. This method first finds a feedback controller that causes the system to be imbued with trajectory robustness, then finds open-loop reference signals that each safely drive the system from a subset of initial states to the goal state. In this paper we use piecewise affine system identification techniques to generate a feedback control law to replace the open-loop signals. This provides additional robustness to unexpected disturbances, in addition to reducing the memory required in the resulting controller, from storing many signals to a set of piecewise affine control laws. Keywords: controller synthesis, feedback controls.**

I. Introduction

The problem of safety/reachability controller synthesis has received a lot of attention from the hybrid systems community. This problem pertains to the task of designing a controller for a (hybrid) dynamical system that guarantees the system states to reach a prescribed goal, without entering an unsafe set.

Following our earlier results [1], [2], [3], we pursue a synthesis strategy that exploits human generated trajectories. The key concept here is to harness valid (in the sense of safety/reachability as described in the previous paragraph) execution trajectories of the system, or the simulations thereof. To generalize the safety property of a simulated execution trajectory to a compact neighborhood around it, we use the concept of trajectory robustness [4], [1] or incremental stability [5], [6]. Roughly speaking, these properties can provide us with a bound on the divergence of the trajectories (i.e. their relative distances in $\mathcal{L}_\infty$). The main conceptual tool that is used in this approach, the *approximate bisimulation*, was developed by Girard and Pappas [7].

The use of approximate bisimulation in safety controller synthesis has also been pursued by other researchers. For example, we refer the readers to the following works by Tabuada, Pola, Girard, Zamani, and Mazo [5], [6], [8], [9], [10]. The notion of approximate bisimulation is used to quantize the continuous state space, which results in a countable transition system approximation of the original dynamics. The controller can then be realized as a symbolic controller. A recent work by Colombo and Del Vecchio [11] also uses similar approach in designing safety controller for differentially flat systems. In our approach, the controller is synthesized using finitely many valid human-generated trajectories [1], [2]. We do not require the open loop dynamics to have incremental stability property. Instead, a part of the controller synthesis procedure is devoted to designing an inner-loop controller (see Figure 1(a)) that establishes this property.

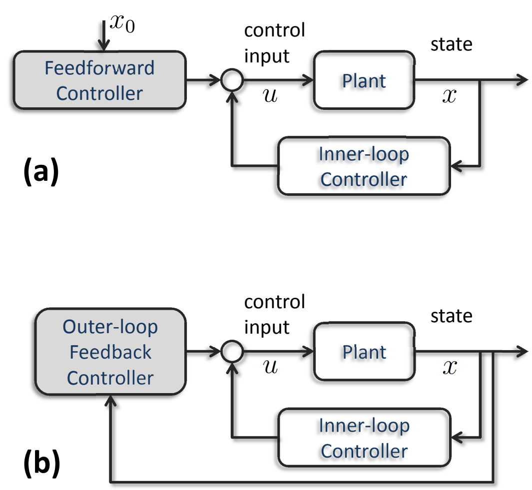


Fig. 1: Block diagrams of our control approach. **(a)** In prior works, the controller consists of two parts, an inner-loop feedback controller and a feedforward controller. **(b)** In this paper, we replace the feedforward controller with an outer-loop feedback controller.

In our previous work, [1], [2] the synthesized controller consists of two parts (see Figure 1(a)). The first part is the inner-loop feedback controller, as explained above. The second part is a feedforward controller that "replays" the appropriate control input trajectories as learned from the human generated trajectories. In this paper, we seek to replace this feedforward controller with a feedback controller. In doing so, we want that (i) the feedback controller is also constructed using the human generated trajectories, and (ii) the validity of the controller in solving the control problem is maintained. Our approach is to learn a piecewise affine control law from the human generated trajectories, following the approach of Bemporad et al [12]. Effectively, the overall control strategy will be a switched control system. The difference between our approach and other related work lies in the fact that we do not need this feedback controller to be valid for the entire state-space. In fact, in keeping with the spirit of our earlier works [1], [2], the feedback controller is only defined on the part of the state-space that has been explored by the humans. At the same time, we also guarantee

Andrew Winn and Agung Julius are with the Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY 12180, Email: winna@rpi.edu, agung@ecse.rpi.edu.

that the designed feedback controller will not steer the state trajectory away from the part that has been explored.

II. Controller Synthesis with Human Generated Trajectories

The discussion in this paper is setup for discrete-time systems, as opposed to our earlier results in [1], [2] that were setup for continuous-time systems. This is because in the subsequent discussion we learn the feedback controller from discrete data.

Consider a discrete-time dynamical system with input

$$\Sigma_{\text{inp}} : x(k+1) = f(x(k), u(k)),\ \ x(\cdot) \in \mathbb{R}^n,\ \ u(\cdot) \in \mathbb{R}^m. \tag{1}$$

Suppose that there is a given compact set of initial states $\text{Init} \subset \mathbb{R}^n$, where the state is initiated at $t = 0$, i.e. $x(0) \in \text{Init}$. Also, we assume that there is a set of goal states, $\text{Goal} \subset \mathbb{R}^n$, and a set of unsafe states $\text{Unsafe} \subset \mathbb{R}^n$. As usual, a trajectory is deemed unsafe if it enters the unsafe set. Suppose that we are given the following control problem:

Problem 1: Design a feedback control law $u = k(x, x_0)$ such that for any initial state $x_0 \in \text{Init}$, the trajectory of the closed loop system enters Goal before time $t = T_{\max}$, and remains safe until it enters Goal.

Hereafter, any trajectory that satisfies the conditions above is called a **valid trajectory**.

The key concept in our approach is the *control autobisimulation function (CAF)*, which was defined in [1] for continuous-time system. Its discrete-time counterpart can be stated as follows.

Definition 1: A continuously differentiable function $\psi : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}_+$ is a control autobisimulation function of (1) if for any $x, x' \in \mathbb{R}^n$,

$$\psi(x, x') \geq \|x - x'\|, \tag{2}$$

and there exists a function $k : \mathbb{R}^n \to \mathbb{R}^m$ such that

$$\psi(f(x, k(x)), f(x', k(x')) \leq \psi(x, x'). \tag{3}$$

The control autobisimulation function is an analog of the control Lyapunov function (CLF) [13], for approximate bisimulation [7], [14]. While control Lyapunov function has been used to construct control laws that guarantee stability (e.g. [15]), we shall use the control autobisimulation function to construct control laws that guarantee trajectory robustness.

If $f(x, u)$ is an affine linear function,

$$f(x, u) \triangleq Ax + Bu + g, \tag{4}$$

we can form the control bisimulation function $\psi(x, x')$ by the quadratic expression

$$\psi(x, x') \triangleq \left[(x - x')^T P (x - x')\right]^{\frac{1}{2}}, \tag{5}$$
$$P^T = P \succeq I, \tag{6}$$

and the control law of the form

$$u = Kx + v, \tag{7}$$

for some reference input signal v of appropriate dimensions. In this case, inequality (3) is equivalent to:

$$(x-x')^T (A+BK)^T P (A+BK)(x-x') \leq (x-x')^T P (x-x'), \tag{8}$$

which is also equivalent to

$$P - (A + BK) P (A + BK)^T \succeq 0. \tag{9}$$

Classical results from Lyapunov stability of discrete-time systems dictate that such K and P exist if and only if (A, B) is stabilizable [16]. Moreover, P and K can be found from the Linear Matrix Inequality [17]:

$$P^T = P \succeq I, \tag{10}$$

$$\begin{bmatrix} P & AP + BD \\ (AP + BD)^T & P \end{bmatrix} \succeq 0, \tag{11}$$

where $D \triangleq KP$ and therefore $K = DP^{-1}$. Observe that this result is independent of the selection of the reference input signal v. To ease the subsequent discussion, we introduce the shorthand notation

$$\tilde{A} \triangleq A + BK, \tag{12}$$

and the following notations.

Notation 1: We denote the solution of

$$x(k+1) = (A + BK)x(k) + Bv(k) + g,$$
$$x(0) = x_0,$$

as $\xi(k; x_0, v)$.

Notation 2: Observing that ψ defines a metric in $\mathbb{R}^n$, we introduce the notation:

$$\|x - x'\|_\psi \triangleq \psi(x, x'), \tag{13}$$

and we define the ball

$$B_\psi(x, r) \triangleq \{x' \mid \|x - x'\|_\psi \leq r\}. \tag{14}$$

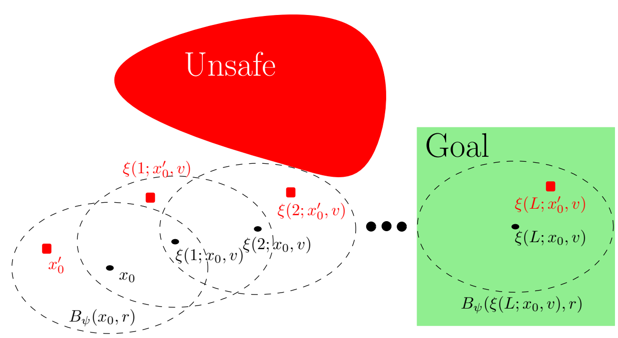


Fig. 2: Illustration of the trajectory robustness property. With a proper signal v, we can steer a neighborhood of initial conditions $B_\psi(x_0, r)$ to satisfy the control objective.

We can directly deduce that for any $x_0' \in \mathbb{R}^n$ and any reference input signal v,

$$\xi(k; x_0', v) \in B_\psi(\xi(k; x_0, v), \psi(x_0, x_0')).$$

This is illustrated in Figure 2. Therefore, if an appropriate reference input signal v can be obtained for the nominal initial state x_0, it can also be used as the input for a ball of initial conditions around x_0 and the resulting trajectories are guaranteed to be valid.

In our approach, the reference input v is obtained from humans playing a simulator of the plant. The overall idea is to obtain multiple reference input signals v for multiple initial conditions, such that the corresponding "robust neighborhoods" cover the set Init. This idea was first presented in [1] and later generalized for nonlinear systems in [2]. The objective of the current paper is to obtain an outer-loop feedback controller that can reproduce the reference input signal v well enough to ensure that the safety criteria are still met.

III. Robustness to Bounded Disturbance

In this section, we study the effect of bounded disturbance in the reference input signal v on the robustness property. That is, we formulate some bound on $\|\xi(k;x_0,v)-\xi(k;x_0',v+\varepsilon)\|_\psi$ in terms of $\|x_0-x_0'\|_\psi$ and $\|\varepsilon\|_\infty$. Here, ε is a disturbance signal affecting the reference input v and $\|\varepsilon\|_\infty$ denotes its ℓ_∞ norm.

Theorem 1: Suppose that $\|\varepsilon\|_\infty \le \delta$. There exists a critical radius $r_{\mathrm{crit}} > 0$ (generally depends on $\|\varepsilon\|_\infty$) such that for any $r > r_{\mathrm{crit}}$ the following is true for all $k > 0$.

$$\|x_0-x_0'\|_\psi \le r \Rightarrow \|\xi(k;x_0,v)-\xi(k;x_0',v+\varepsilon)\|_\psi \le r. \tag{15}$$

Proof: We will prove this result by induction. First, from the relationship

$$\xi(1;x_0,v) = \tilde{A}x_0 + Bv(0) + g, \tag{16}$$

$$\xi(1;x_0',v+\varepsilon) = \tilde{A}x_0' + Bv(0) + B\varepsilon(0) + g, \tag{17}$$

it can be shown that

$$\begin{aligned}\|\xi(1;x_0,v)-\xi(1;x_0',v+\varepsilon)\|_\psi^2 - \|x_0-x_0'\|_\psi^2 = \\ (x_0-x_0')^T\left(\tilde{A}^T P\tilde{A}-P\right)(x_0-x_0') \\ -(x_0-x_0')^T\tilde{A}^T PB\varepsilon(0) \\ -\varepsilon^T(0)B^T P\tilde{A}(x_0-x_0') \\ +\varepsilon^T(0)B^T PB\varepsilon(0).\end{aligned} \tag{18}$$

By design, we know that $\left(P-\tilde{A}^T P\tilde{A}\right)$ is positive definite (see (9)). Denote the smallest singular value of $\left(P-\tilde{A}^T P\tilde{A}\right)$ as $\sigma_{\min}$. We can majorize the first (quadratic) term by $-\sigma_{\min}\|x_0-x_0'\|^2$, the middle (linear) terms by $2\|x_0-x_0'\|\left\|\tilde{A}^T PB\right\|\delta$, and the final (constant) term by $\left\|B^T PB\right\|\delta^2$, yielding

$$\begin{aligned}\|\xi(1;x_0,v)-\xi(1;x_0',v+\varepsilon)\|_\psi^2 - \|x_0-x_0'\|_\psi^2 \le \\ -\sigma_{\min}\|x_0-x_0'\|^2 + 2\|x_0-x_0'\|\left\|\tilde{A}^T PB\right\|\delta + \left\|B^T PB\right\|\delta^2.\end{aligned} \tag{19}$$

A sketch of the plot of the right hand side of (19) versus $\|x_0-x_0'\|$ can be seen in Figure 3.

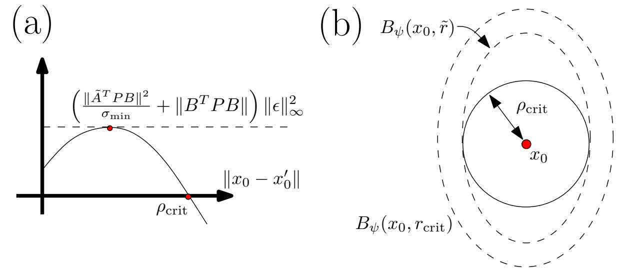


Fig. 3: (a) A sketch of the right hand side of (19) as a function of $\|x_0-x_0'\|$. (b) An illustration relating ρ_{crit}, $\tilde{r}$, and r_{crit}.

Observe that if

$$\|x_0-x_0'\| \ge \underbrace{\frac{\left\|\tilde{A}^T PB\right\| + \sqrt{\left\|\tilde{A}^T PB\right\|^2 + \sigma_{\min}\left\|B^T PB\right\|}}{\sigma_{\min}}\delta}_{\triangleq \rho_{\mathrm{crit}}}, \tag{20}$$

then

$$\|\xi(1;x_0,v)-\xi(1;x_0',v+\varepsilon)\|_\psi^2 \le \|x_0-x_0'\|_\psi^2. \tag{21}$$

Now, define

$$\tilde{r} \triangleq \sqrt{\lambda_{\max}}\rho_{\mathrm{crit}}, \tag{22}$$

where $\lambda_{\max}$ is the largest eigenvalue of P, and therefore

$$\|x_0-x_0'\|_\psi \ge \tilde{r} \Rightarrow \|x_0-x_0'\| \ge \rho_{\mathrm{crit}}. \tag{23}$$

We can then define

$$r_{\mathrm{crit}} = \sqrt{\tilde{r}^2 + \left(\frac{\left\|\tilde{A}^T PB\right\|^2}{\sigma_{\min}} + \left\|B^T PB\right\|\right)\delta^2}. \tag{24}$$

We can infer that for any $r \ge r_{\mathrm{crit}}$,

$$\|x_0-x_0'\|_\psi \le r \Rightarrow \|\xi(1;x_0,v)-\xi(1;x_0',v+\varepsilon)\|_\psi \le r. \tag{25}$$

Since the system is time-invariant, we can repeat this reasoning to obtain (15). ■

Remark 2: As $\sigma_{\min}$ becomes arbitrarily small, so too must δ, which reduces the number of datapoints that can be fit by a single linear affine map. We can ensure a minimum on the singular value if instead of (10), we solve

$$\begin{bmatrix} P & AP+BD \\ (AP+BD)^T & P-\sigma_{\min}I \end{bmatrix} \succeq 0,$$

IV. Feedback Controller Learning Algorithm

In this section we develop a method to generate a piecewise affine feedback controller $K_v(x)$, that is,

$$K_v(x) = \begin{cases} \theta_1 x + \theta_{0,1}, & \text{when } x \in \mathcal{X}_1 \\ \theta_2 x + \theta_{0,2}, & \text{when } x \in \mathcal{X}_2 \\ \vdots & \vdots \\ \theta_n x + \theta_{0,n}, & \text{when } x \in \mathcal{X}_n \end{cases}, \tag{26}$$

where the $\{\mathcal{X}_j\}_{j=1}^n$ form a partition of the state space such that when $K_v(x)$ replaces v in (7), the feedback control law satisfies the safety controller synthesis problem for the states covered by the robustness balls in the feedforward control law. In this section we shall make use of the following notations.

Notation 3: Given a feedforward controller consisting of N nominal trajectories, the subscript i shall denote the variables corresponding to the ith nominal trajectory, that is, $x_{0,i}$, $v_i(k)$, r_i, and T_i represent the initial state, the reference input signal, the robustness, and the termination time of the ith trajectory, respectively.

Notation 4: For brevity we denote the ith nominal trajectory as

$$\xi_i(k) \triangleq \xi(k; x_{0,i}, v_i). \tag{27}$$

Notation 5: We define the robustness neighborhoods around the nominal trajectories as $\mathcal{F}$. Formally,

$$\mathcal{F} \triangleq \bigcup_{i=1}^{N} \bigcup_{k=0}^{T_i} \{x \mid x \in B_\psi(\xi_i(k), r_i)\}. \tag{28}$$

Intuitively, F represents the part of the state space that has been explored by the human players.

The following results will be useful in developing our feedback law. Their proofs are omitted due to space contraints.

Theorem 3: Given a controller with feasible states $\mathcal{F}$, if

$$x(\ell) \in \mathcal{F} \setminus \mathrm{Goal} \;\Rightarrow\; x(\ell+1) \in \mathcal{F} \tag{29}$$

then the controller is safe (in the sense of Problem 1) for all trajectories that begin in Init and terminate upon entering Goal.

The following theorem imposes restraints on the partitions of our piecewise affine feedback controller to reduce (29) to a more tractable condition.

Theorem 4: Let the partitions in the feedback controller be chosen such that if $x \in \mathcal{F} \cap \mathcal{X}_j$ then there exists a $\xi_i(k) \in \mathcal{X}_j$ such that

$$x \in B_\psi(\xi_i(k), r_i). \tag{30}$$

In this case, condition (29) of Theorem 3 will hold if for all $x \in B_\psi(\xi_i(k), r_i)$ where $\xi_i(k) \in \mathcal{X}_j$,

$$(A + BK)x + B\theta_j x + B\theta_{0,j} + g \in B_\psi(\xi_i(k+1), r_i), \tag{31}$$

that is, the feedback law at state $\xi_i(k)$ maps the robustness ball around $\xi_i(k)$ into the robustness ball around $\xi_i(k+1)$. In Section V we will examine how to generate partitions such that the conditions of Theorem 4 are met.

Remark 5: In the remainder of this analysis, we will assume our feedback control law is partitioned in a way that satisfies the requirements of Theorem 4. Therefore, when we say $x \in B_\psi(\xi_i(k), r_i)$, we can assume $x \in \mathcal{F}$ and take $\xi_i(k)$ to be the point such that both it and x are in the same partition, since an $\xi_i(k)$ satisfying this must exist for each $x \in \mathcal{F}$. Note that if $x \in B_\psi(\xi_i(k), r_i)$ then $x \in \mathcal{F}$ by definition.

Assume for the moment that our controller is exact for each $\xi_i(k)$ within a given partition, so that

$$v_i(k) = \theta_j \xi_i(k) + \theta_{0,j}.$$

Consider a trajectory of our feedback system such that $\xi(\ell; x_0', K_v) \in B_\psi((\xi_i(k), r_i)$. We see that

$$\begin{aligned}\xi_i(k+1) &- \xi(\ell+1; x_0', K_v)\\ &= (\tilde{A} + B\theta_j)(\xi_i(k) - \xi(\ell; x_0', K_v))\end{aligned}$$

If inequality (3) is satisfied then $\xi(\ell; x_0', K_v) \in B_\psi(\xi_i(k), r_i)$ and the condition of Theorem 4 will be satisfied. Using the analysis in Section II we see that (3) will hold if

$$\begin{bmatrix} P & \tilde{A}P + B\theta_j P \\ (\tilde{A}P + B\theta_j P)^T & P \end{bmatrix} \succeq 0. \tag{32}$$

Now, Let us relax the requirement that the affine map is exact. In this case

$$v_i(k) = \theta_j \xi_i(k) + \theta_{0,j} - \varepsilon_i(k).$$

Let

$$\tilde{\tilde{A}} = \tilde{A} + B\theta_j = A + BK + B\theta_j.$$

If (32) is satisfied, then the results in Section III hold with $\tilde{\tilde{A}}$ substituted for $\tilde{A}$; let $\tilde{r}_{\mathrm{crit}}$ be the solution to (24) under this substitution. In this case, if

$$\tilde{r}_{\mathrm{crit}} < r_i, \text{ for all trajectories} \tag{33}$$

then $\xi(\ell; x_0', K_v) \in B_\psi(\xi_i(k), r_i)$ and the condition of Theorem 4 will be satisfied.

This inspires the following algorithm adapted from Bemporad et al. [12] for learning the feedback control. Let $(v_i(k), \xi_i(k))$ for all i and k be the data for which we are going to learn a piecewise affine mapping. The basic idea is to find affine maps that cover the largest number of data points and guarantee safety. Once a set of point clusters are found, we will use the method outlined in Section V to generate a partition for the clusters.

We start by finding a map using all points, then iterate the process over the remaining unmapped points. If we come across a mapping that covers more points than a previous map, we discard the previous map and all subsequent maps and replace it with the current map. This in general will lead to fewer partitions [12]. To find each map a search algorithm with thermal relaxation is employed to search the parameter space for $\theta_j, \theta_{0,j}$. Then, the map that satisfies the most data points that is found over the relaxation process is returned. See [12] for details.

Our method modifies the algorithm presented in [12] in several ways:

1) δ is chosen so that if

$$|\theta_j \xi_i(k) + \theta_{0,j} - v_i(k)| < \delta,$$

then $r_{\mathrm{crit}} < r_i$ for all trajectories, using (20)–(22). Note that there is a trade-off in how we choose δ: too large and an arbitrarily small θ_j can cause $\tilde{r}_{\mathrm{crit}} < r_i$ to be violated for some i; too small, and the affine map will only be able to map very few points.

2) For each candidate θ_j that fits more points than the current best candidate, we verify that (32) and (33) hold.
3) We initialize the algorithm with θ_j as the zero row vector (or matrix in the multidimensional case) and set $\theta_0 = v_i(k)$ for some (i, k) that corresponds to an unmapped data point. We see that K was designed such that (33) holds for $\tilde{\tilde{A}} = \tilde{A} + 0 = A + BK$, and that $|\theta_0 \xi_i(k) - v_i(K)| = 0 < \delta$. This guarantees that at least one point is mapped at each iteration, and therefore that the process will eventually terminate.

V. Designing Feedback Controller Partitions

The method outlined in Section IV will generate a set of datapoint clusters and corresponding affine feedback laws that guarantee safety over the robustness balls around each point in the cluster. We split the clusters into subclusters such that for any two subclusters, the convex hulls of both subclusters do not intersect. We can now use techniques such as support vector machines [18] or robust linear programming [19] to find separating hyperplanes for each cluster.

These hyperplanes define a polyhedral partition of the state space. Let $\hat{\mathcal{X}}_j$ denote the jth partition and let J denote the total number of partitions. Assume $\hat{\mathcal{X}}_j$ has M faces and is defined by

$$a_m^T x - b_m \leq 0, \qquad m = 1, 2, \ldots, M. \tag{34}$$

The robustness balls about the points in each partition now have a feedback control law that guarantees the safe exectution of the system. However, it is possible that a system state within one partion may belong to a robustness ball in another partition and not a robustness ball within the same partition. In this case Theorem 4 is not satisfied and safety cannot be guaranteed. This is demonstrated by point x in Figure 4b. To handle these cases we must manipulate the partition boundaries.

Let

$$I_m \triangleq \left\{ (i,k) \;\middle|\; 0 < a_m^T \xi_i(k) - b_m < \left\| a_m^T P^{-\frac{1}{2}} \right\| r_i \right\},$$
$$b_m^- \triangleq \min_{(i,k) \in I_m} a^T \xi_i(k) - \left\| a_m^T P^{-\frac{1}{2}} \right\| r_i.$$

I_m is the set of indexes of datapoints outside of $\hat{\mathcal{X}}_j$ whose robustness balls intersect the mth face of $\hat{\mathcal{X}}_j$, and b_m^- is defined such that no datapoint outside of the hyperplane defined by the mth face has a robustness ball that intersects the hyperplane defined by

$$a_m^T x - b_m^- \leq 0$$

Refer to Figure 4 for a graphical representation.

Our goal is to infer a new partition $\{\mathcal{X}_j\}_{j=1}^J$ from $\{\hat{\mathcal{X}}_j\}_{j=1}^J$ such that the assumption of Theorem 4 hold. An algorithm for generating a new partition with these properties is given in Figure 5.

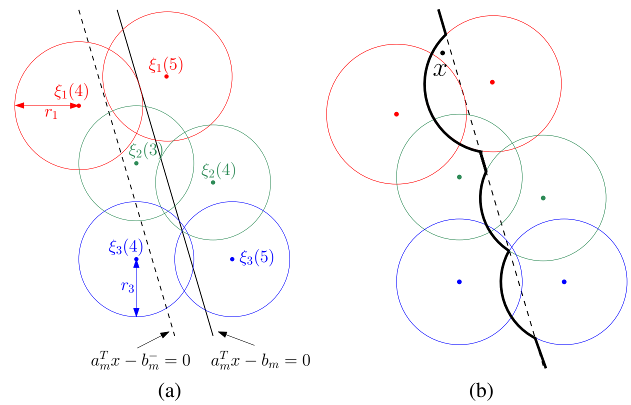


Fig. 4: (a) Graphical representation of I_m and b_m^-. Here, $P = I$, the identity matrix, so that the robustness balls are represented by circles of radius i for simplicity. In this case the ball around $\xi_1(5)$ determines b_m^-. (b) Feedback controller partition as implemented by algorithm in Figure 5. Note that x is now assigned to the partition that will guarantee safety.

Given: x, $\{\hat{\mathcal{X}}_j\}_{j=1}^J$
if $a_m^T x - b_m^- \leq 0$ **then**
  return $\hat{\mathcal{X}}_j$
else
  Let $m_n, n = 1, \ldots, N$ denote indexes where $a_m^T x - b_m^- > 0$
  for $n = 1, 2, \ldots, N$ **do**
    Let $\hat{\mathcal{X}}_{j_m}$ denote the partition that shares face m with $\hat{\mathcal{X}}_j$
    for $(i, k) \in I_{m_n}$ **do**
      if $x \in B_\psi(\xi_i(k), r_i)$ **then**
        return $\hat{\mathcal{X}}_{j_{m_n}}$
      end if
    end for
  end for
  return $\hat{\mathcal{X}}_j$
end if

Fig. 5: An algorithm for generating feedback controller partitions that maintain safety properties of controller

VI. Numerical Example

In this section we apply our method to a feedforward controller generated for the linear affine system and safety controller synthesis problem considered in our earlier work [1], discretized with a sampling rate of $f_s = 0.1s$. This system is given by

$$\Sigma : \dot{x} = Ax + Bu + g$$

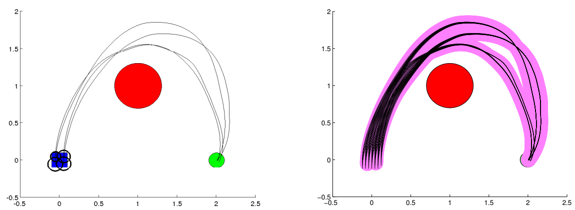
where

$$A = \begin{bmatrix} 0.999 & 0.01 & 0 & 0 \\ -0.01 & 1.001 & 0 & 0 \\ 0 & 0 & 0.999 & 0.01 \\ 0 & 0 & -0.01 & 1.001 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0.01 & 0 \\ 0 & 0.01 \\ 0 & 0 \end{bmatrix} \quad g = \begin{bmatrix} 0.01 \\ 0 \\ 0.01 \\ 0 \end{bmatrix}$$

The three sets defining the safety controller synthesis problem considered are given by

$$\text{Init} = \left\{ x \in \mathbb{R}^4 \;\middle|\; |x_1| \leq 0.2, |x_3| \leq 0.2, x_2 = 0, x_4 = 1 \right\}$$
$$\text{Unsafe} = \left\{ x \in \mathbb{R}^4 \;\middle|\; \sqrt{(x_1 - 1)^2 + (x_2 - 1)^2} \leq 0.3 \right\}$$
$$\text{Goal} = \left\{ x \in \mathbb{R}^4 \;\middle|\; \sqrt{(x_1 - 2)^2 + x_2^2} \leq 0.1 \right\}$$

A feedforward controller was generated for this system using human trajectories via the method presented in [1]. This controller consists of four trajectories and a total of 1564 input reference signal values. These four trajectories and the corresponding coverage of Init are shown in Figure 6a.



(a) (b)

Fig. 6: (a) Trajectories used in feedforward controller for this safety controller synthesis problem. The rectangle on the left is Init, the circle in the upper middle is Unsafe, and the circle on the right is Goal. (b) Trajectories generated by feedback controller over a sampling of initial conditions. The background region represents $\mathcal{F}$, the region over which the controller is valid.

At this point we applied the presented method for generated a feedback controller. We solved for P and K, requiring that $\sigma_{\min}(P - \tilde{A}^T P \tilde{A}) > 0.1$, which yielded

$$P = \begin{bmatrix} 7.03 & -5.17 & 0 & 0 \\ -5.17 & 6.85 & 0 & 0 \\ 0 & 0 & 7.03 & -5.17 \\ 0 & 0 & -5.17 & 6.85 \end{bmatrix} \quad K = -\begin{bmatrix} 2.34 & 3.38 & 0 & 0 \\ 0 & 0 & 2.37 & 3.38 \end{bmatrix}.$$

We then designed δ such that $r_{\text{crit}} < \min_i r_i/2$, that is, we chose the critical radius to be half of the smallest robustness radius. We then ran the algorithm and it was able to produce a linear affine feedback controller that split the 1564 data points into 110 partitions. This controller was then applied to a sampling of points to generate the trajectories shown in Figure 6b.

VII. Conclusion

In this paper, we first presented a method for generating an initial feedback controller that imbues a system trajectory robustness for discrete time systems. We next considered conditions for ensuring trajectory robustness in the presence of a uniformly bounded disturbance at the input. We then gave sufficient conditions for designing a piecewise feedback control law from a feedforward controller that transfers safety from the latter to the former. Following this we presented a method for designing a piecewise affine feedback control law that guarantees safety. We then addressed the issue of defining the partitions of the piecewise law to meet the sufficient conditions for maintaining safety. Finally, we presented a numerical example wherein this method was used to generate a feedback control law from a human-generated feedforward control law, and controller was verified over a sampling of initial states.

VIII. Acknowledgements

The research reported in this paper is partially supported by the NSF through the grants CAREER CNS-0953976 and CNS-1218109, and the Department of Defense SMART Scholarship.

References

[1] A. A. Julius and S. Afshari, "Using computer games for hybrid systems controller synthesis," in *Proc. 49th IEEE Conf. Decision and Control*, Atlanta, Georgia, 2010, pp. 5887–5892.

[2] A. A. Julius and A. K. Winn, "Safety controller synthesis using human generated trajectories: Nonlinear dynamics with feedback linearization and differential flatness," in *Proc. American Control Conference*, Montreal, Canada., 2012, pp. 709–714.

[3] A. K. Winn and A. A. Julius, "Optimization of human generated trajectories for safety controller synthesis," in *to appear in Proc. American Control Conference*, Washington DC, USA., 2013.

[4] A. A. Julius, "Trajectory-based controller design for hybrid systems with affine continuous dynamics," in *Proc. IEEE Conf. Automation Science and Engineering*, Toronto, Canada, 2010, pp. 1007–1012.

[5] P. Tabuada, "An approximate simulation approach to symbolic control," *IEEE Trans. Automatic Control*, vol. 53, no. 6, pp. 1406–1418, 2008.

[6] G. Pola, A. Girard, and P. Tabuada, "Approximately bisimilar symbolic models for nonlinear control systems," *Automatica*, vol. 44, no. 10, pp. 2508–2516, 2008.

[7] A. Girard and G. J. Pappas, "Approximation metrics for discrete and continuous systems," *IEEE Trans. Automatic Control*, vol. 52, no. 5, pp. 782–798, 2007.

[8] J. Camara, A. Girard, and G. Goessler, "Safety controller synthesis for switched systems using multi-scale symbolic models," in *Proc. IEEE Conf. Decision and Control*, Orlando, Florida, 2011.

[9] M. Zamani and P. Tabuada, "Towards backstepping design for incremental stability," *IEEE Trans. Automatic Control*, vol. 56, no. 9, 2011.

[10] M. Zamani, G. Pola, M. Mazo Jr., and P. Tabuada, "Symbolic models for nonlinear control systems without stability assumptions," *IEEE Trans. Automatic Control*, vol. 57, no. 7, pp. 1804–1809, 2012.

[11] A. Colombo and D. Del Vecchio, "Supervisory control of differentially flat systems based on abstraction," in *Proc. IEEE Conf. Decision and Control*, Orlando, Florida., 2011.

[12] A. Bemporad, A. Garulli, S. Paoletti, and A.Vicino, "A bounded-error approach to piecewise affine system identification," *IEEE Trans. Automatic Control*, vol. 50, no. 10, pp. 1567–1580, 2005.

[13] Z. Artstein, "Stabilization with relaxed controls," *Nonlinear Analysis*, vol. 15, no. 11, pp. 1163–1170, 1983.

[14] A. A. Julius, G. Fainekos, M. Anand, I. Lee, and G. J. Pappas, "Robust test generation and coverage for hybrid systems," in *Hybrid Systems: Computation and Control*, ser. LNCS, vol. 4416. Springer Verlag, 2007, pp. 329–342.

[15] E. D. Sontag, "A 'universal' construction of Artstein's theorem on nonlinear stabilization," *Systems and Control Letters*, vol. 13, no. 2, pp. 117–123, 1989.

[16] K. Ogata, *Discrete-Time Control Systems*. Prentice Hall, 1995.

[17] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in Systems and Control Theory*. Philadelphia: SIAM, 1994.

[18] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, pp. 273–297, 1995. [Online]. Available: http://dx.doi.org/10.1007/BF00994018

[19] K. P. Bennett and O. L. Mangasarian, "Robust linear programming discrimination of two linearly inseparable sets," *Optimization Methods and Software*, vol. 1, no. 1, pp. 23–34, 1992. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1080/10556789208805504