2016 IEEE Conference on Computer Aided Control System Design (CACSD)
Part of 2016 IEEE Multi-Conference on Systems and Control
September 19-22, 2016. Buenos Aires, Argentina

Control of Discrete-Time Piecewise Affine Probabilistic Systems using Reachability Analysis

Leonhard Asselborn[1] and Olaf Stursberg[1]

Abstract— **This paper proposes an algorithmic approach to synthesize stabilizing control laws for discrete-time piecewise affine probabilistic (PWAP) systems based on computations of probabilistic reachable sets. The considered class of systems contains probabilistic components (with Gaussian distribution) modeling additive disturbances and state initialization. The probabilistic reachable state sets contain all states that are reachable with a given confidence level under the effect of time-variant control laws. The control synthesis uses principles of the ellipsoidal calculus, and it considers that the system parametrization depends on the partition of the state space. The proposed algorithm uses LMI-constrained semi-definite programming (SDP) problems to compute stabilizing controllers, while polytopic input constraints and transitions between regions of the state space are considered. The formulation of the SDP is adopted from a previous work in [1] for switched systems, in which the switching of the continuous dynamics is triggered by a discrete input variable. Here, as opposed to [1], the switching occurs autonomously and an algorithmic procedure is suggested to synthesis a stabilizing controller. An example for illustration is included.**

I. Introduction

This paper addresses the task of controlling discrete-time piecewise affine probabilistic (PWAP) systems, which consist of a partition of the state space and a collection of affine dynamics valid in each region. Probabilistic uncertainties with respect to the initial state and additive disturbances are considered with Gaussian distribution. In general, piecewise affine (PWA) systems are convenient mathematical models for practical application, since discontinuities arising from saturation constraints, hysteresis, or friction can be encoded. Furthermore, PWA systems enable to encode linearizations of originally nonlinear dynamics for a finite number of state space regions [2]. The focus of this contribution is to provide stabilizing time-variant state feedback control laws for set-to-set transitions of the system state, while ensuring a given probability level. As a motivating example, consider the translational dynamics of a ship in the open sea, and the task of reaching a given target set. Suppose that the open sea is divided into different regions, where the current of the sea flows in different directions. Hence, the dynamics of a ship is described by different sets of nonlinear differential equation in each region. A linearization of the nonlinear dynamics in each region of the sea yields a PWA system, and the proposed procedure in this contribution provides a control law to reach the target set. This law takes the different flows in each region of the sea into account, as well as it ensures a specified probability bound for reaching the target.

PWAP systems are a special class of stochastic hybrid systems, and a considerable part of research on verification and control of hybrid systems is devoted to reachability computations. Whereas the question of whether a target set is reached from an initial set is undecidable in the general case, algorithmic reachable set computation has gained significant attention in the last two decades. An important class of methods is to determine conservative over-approximations of reachable sets in form of polytopes ([3], [4], [5]), zonotopes ([6], [7]), or support functions ([8]). These variants differ in the compromise between accuracy, computational effort, and memory requirement for obtaining the over-approximations. For PWA systems, [9], [10] presented a method for controller synthesis based on problem decomposition: first, the continuous reachability problem is solved on sets of simplices such that any simplex is left through an exit facet. Secondly, a discrete control problem is solved to obtain a feasible discrete state path. For the same system class with bounded disturbances, an optimal control problem is addressed by dynamic programming in [11]; this approach is extended to state- and input-dependent disturbances in [12]. The work by [13] provides a technique for controller synthesis based on a discrete abstraction of the PWA that takes into account the control inputs.

As predecessor of the work to be presented here, [14] introduced a method for synthesizing controllers for nonlinear discrete-time systems with bounded disturbances in a non-stochastic setting. The main idea is to solve semi-definite programming (SDP) problems formulated for conservative linearizations of the dynamics in order to obtain robust time-variant control laws for set-to-set control. The concept was extended to a stochastic setting for discrete-time linear systems in [15]. The underlying principle was to synthesize feedback control laws by SDP such that the controlled system is stabilized to a given confidence level. The state sets, which are reachable under the effect of a synthesized continuous control law to a specified (high) probability, are represented by ellipsoids. This choice is suitable with respect to the compatibility with the synthesis procedure using SDP. The task of probabilistic stabilization considers polytopic input constraints. This work is extended in [16] to switched linear dynamics, in which the current active dynamic is determined by a discrete input. In order to select an appropriate discrete input in any time step, a type of tree search is adopted, which steers the system evolution towards a given target set. The techniques developed for switched linear systems

[1]Control and System Theory, Dept. of Electrical Engineering and Computer Science, University of Kassel (Germany). Email: {l.asselborn, stursberg}@uni-kassel.de

978-1-5090-0758-5/16/$31.00 ©2016 IEEE

are extended in the present paper to PWAP, for which the switching occurs autonomously, and is thus not available as a degree of freedom for control.

The paper first introduces the class of systems and control problem (Sec. II), and Sec. III specifies the control law and the SDP. Reachability analysis for PWAP systems is covered in Sec. IV, and the optimization-based solution procedure is proposed in Sec. V. Numerical results for an example are provided in Sec. VI, before Sec. VII concludes the paper.

II. System and Problem Definition

A. Preliminaries

This section first clarifies the notation and recalls some basic definitions and facts used throughout the paper.

Let $\mathcal{E}$ denote the set of all ellipsoidal sets in $\mathbb{R}^n$. An *ellipsoidal set* $\varepsilon(q, Q) \in \mathcal{E}$ is parametrized by a center point $q \in \mathbb{R}^n$ and a symmetric and positive-definite shape matrix $Q \in \mathbb{R}^{n\times n}$ according to:

$$\varepsilon(q, Q) = \left\{x \in \mathbb{R}^n | (x-q)^T Q^{-1}(x-q) \leq 1\right\} \quad (1)$$

with ^{T} indicating the transpose of the vector.

The semi-major axis is the largest radius of an ellipsoid, and the semi-minor axis is the smallest radius of an ellipsoid. If $\Lambda(Q)$ denotes the set of eigenvalues of the matrix Q, the length and orientation of the semi-axes of $\varepsilon(q, Q)$ can be described by the root of the eigenvalues $\lambda_i \in \Lambda(Q)$ and the eigenvectors v_i, respectively.

Let $\mathcal{P}$ denote the set of all polytopic sets in $\mathbb{R}^n$. A *convex polytope* $P \in \mathcal{P}$ is the intersection of n_p halfspaces, such that $P = \{x \in \mathbb{R}^n \mid Rx \leq b, R \in \mathbb{R}^{n_p\times n}, b \in \mathbb{R}^{n_p}\}$. Each halfspace is defined by a tuple (r_i, b_i), with $||r_i|| = 1$.

The distance of a point q to the half plane $r_i x = b_i$ is given by:

$$d\left((r_i, b_i), q\right) := r_i q - b_i \quad (2)$$

The interior and boundary of a set $\Theta \subset \mathbb{R}^n$ is denoted by $int(\Theta)$ and $\partial(\Theta)$, respectively, and it holds that:

$$int(\Theta) \cup \partial(\Theta) = \Theta. \quad (3)$$

The multivariate normal distribution of an n-dimensional random vector ξ with covariance matrix $Q_\xi \in \mathbb{R}^{n\times n}$, $Q_\xi = Q_\xi^T \geq 0$, and mean vector $q_\xi \in \mathbb{R}^n$ is denoted by:

$$\xi \sim \mathcal{N}(q_\xi,\ Q_\xi). \quad (4)$$

The probability density function of a multivariable normal distribution has surfaces of equal density, which are described by ellipsoids. The shapes of this ellipsoids are determined by the covariance matrix Q_ξ, and a value c_ξ (see [17]):

$$(\xi - q_\xi)^T Q_\xi^{-1}(\xi - q_\xi) = c_\xi. \quad (5)$$

The ellipsoid $W \in \mathcal{E}$ containing the realizations of the random variable ξ for a given c_ξ is given by:

$$W = \varepsilon(q_\xi, Q_\xi c_\xi) \in \mathcal{E}.$$

In [16], it has been shown that $W^\delta = \varepsilon(q_\xi, Q_\xi c_\xi)$ is a confidence ellipsoid with a scaling factor c_ξ. The latter is computed from the cumulative distribution function of a χ^2-distribution, and $Pr(\xi \in W^\delta) = \delta$ holds.

The sum of two Gaussian variables $\xi_1 \sim \mathcal{N}(q_{\xi_1}, Q_{\xi_1})$ and $\xi_2 \sim \mathcal{N}(q_{\xi_2}, Q_{\xi_2})$ is again a Gaussian variable with the following distribution:

$$\xi_1 + \xi_2 \sim \mathcal{N}(q_{\xi_1} + q_{\xi_2}, Q_{\xi_1} + Q_{\xi_2}). \quad (6)$$

B. System Definition

The class of systems under consideration are discrete-time PWAP systems with input constraints and uncertain state initialization. Their dynamics is given by the composition of the dynamics for the continuous state $x_k \in \mathbb{R}^n$ and the discrete state $z_k \in Z := \{1, \ldots, n_z\}$ with time index k. The continuous state space $\mathbb{R}^n$ is partitioned into n_z polytopic regions $\Theta^{(i)} \in \mathcal{P}$, $i \in Z$ defined by $R_x^{(i)} \in \mathbb{R}^{n_{x,i}\times n}$ and $b_x^{(i)} \in \mathbb{R}^{n_{x,i}}$. For any pair (i, j) with $i \in Z$, $j \in Z$, and $i \neq j$ it holds that:

$$int\left(\Theta^{(i)}\right) \bigcap int\left(\Theta^{(j)}\right) = \emptyset, \text{ and } \mathbb{R}^n = \bigcup_{i=1}^{n_z} \Theta^{(i)}, \quad (7)$$

i.e. the interior of the regions do not overlap, and the union of the regions yields the continuous state space. To any region $\Theta^{(i)} \in \bar{\Theta} := \{\Theta^{(1)}, \ldots, \Theta^{(n_z)}\}$ a (possibly) different dynamics for x_k is assigned. The discrete state z_k encodes the index i of the current region which and determines the continuous dynamics active in time k:

$$x_{k+1} = A_{z_k} x_k + B_{z_k} u_k + G_{z_k} v_k. \quad (8)$$

Here, $u_k \in \mathbb{R}^m$ is the continuous input, and $v_k \in \mathbb{R}^n$ the disturbance vector.

The initial state x_0 is assumed to be Gaussian distributed, i.e. for a mean $q_{x,0} \in \mathbb{R}^n$ and a covariance matrix $Q_{x,0} \in \mathbb{R}^{n\times n}$ with $Q_{x,0} = Q_{x,0}^T \geq 0$ it applies:

$$x_0 \sim \mathcal{N}(q_{x,0}, Q_{x,0}) \quad (9)$$

Likewise, the disturbance input v_k is a normal random variable with mean $q_v \in \mathbb{R}^n$ and covariance matrix $Q_v \in \mathbb{R}^{n\times n}$ and $Q_v = Q_v^T \geq 0$:

$$v_k \sim \mathcal{N}(q_v, Q_v). \quad (10)$$

Its effect on the dynamics is scaled by $G_{z_k} \in \mathbb{R}^{n\times n}$.

The continuous input is constrained to a polytope $U \in \mathcal{P}$, which is parametrized by $R_u \in \mathbb{R}^{n_u\times m}$ and $b_u \in \mathbb{R}^{n_u}$:

$$u_k \in U = \{u_k \mid R_u u_k \leq b_u\}. \quad (11)$$

To prepare the formulation of a feasible execution of the PWAPS, let:

$$getAdjacentReg(x_k)\ :\ \mathbb{R}^n \to 2^Z \quad (12)$$

denote a function to determine the regions in which x_k is contained: if $x_k \in int(\Theta^{(i)})$, the result of (12) is i, while for $x_k \in \partial(\Theta^{(i)})$, the result is the subset of Z referring to discrete states which share a common boundary $\partial\left(\Theta^{(i)}\right)$ in x_k. An admissible execution of the PWAP system is as follows:

Definition 2.1: Let (x_0, z_0) be an initialization with $x_0 \in \Theta^{(z_0)}$. A sequence of pairs (x_k, z_k), $k \in \{0, 1, \ldots\}$ is called *admissible* for the PWAP system, if, for any k, the pair (x_{k+1}, z_{k+1}) is determined from (x_k, z_k) by the following order of computations:

1) sample the disturbance $v_k \sim \mathcal{N}(q_v, Q_v)$
2) choose a feasible input $u_k \in U$
3) compute x_{k+1} according to (8) for $(A_{z_k}, B_{z_k}, G_{z_k})$
4) determine z_{k+1} according to the rule:
   if $x_{k+1} \in int(\Theta^{(i)})$, $i \in Z$ **do** $z_{k+1} := i$
   elseif $x_{k+1} \in \partial(\Theta^{(i)})$, $i \in Z$ **do**
   $Z_{adj} := getAdjacentReg(x_{k+1})$
   if $z_k \in Z_{adj,k+1}$ **do** $z_{k+1} := z_k$
   else $z_{k+1} := \min_{z \in Z_{adj,k+1}} z$
   end
   end

$\triangle$

Assumption 2.1: For any $\Theta^{(i)} \in \bar{\Theta}$ with $0 \in \Theta^{(i)}$, let the origin $0 \in \mathbb{R}^n$ determine an equilibrium point of (8) with $z_k = i$.

C. Problem Definition

As mentioned in Sec. II-A, it is possible to derive a confidence ellipsoid for an n-dimensional random variable. Thus, the initial confidence ellipsoid of a PWAP system is introduced as:

$$X_0^\delta := \varepsilon(q_{x,0}, Q_{x,0} c_x). \tag{13}$$

It has been shown in [1], that the evolution of x_k according to (8) yields a random variable in each time step. Therefore, it is possible to provide a confidence ellipsoid $X_k^\delta = \varepsilon(q_{x,k}, Q_{x,k} c_x)$ for each k.

Problem 2.1: Let a PWAP system as defined before, a terminal region $\mathbb{T} = \varepsilon(0, Q_T) \subset \mathbb{R}^n$ centered in the origin $0 \in \mathbb{R}^n$, and an initial confidence set of states X_0^δ be given. Find a control law:

$$u_k = \kappa_k(x_k), \quad x_k \in X_k^\delta, \tag{14}$$

which transfers the initial state $x_0 \in X_0^\delta$ into $\mathbb{T}$ with probability δ after N steps. $\square$

It is assumed, that there exists a terminal controller, which renders $\mathbb{T}$ probabilistically invariant once X_N^δ is contained in $\mathbb{T}$, i.e. the state x is held in $\mathbb{T}$ with probability δ for $k > N$.

III. Control law specification

The objective is to develop an algorithmic method to solve problem 2.1. To specify the structure of the control law (14), a local time-variant, continuous, affine state feedback controller of the following form is selected:

$$u_k = \kappa_k(x_k) = -K_k x_k + d_k \in U, \ \forall \ x_k \in X_k^\delta. \tag{15}$$

Thus, a solution of Problem 2.1 is established by a set of control tuples $(K_k, d_k) \ \forall \ k \in \{0, 1, \ldots, N-1\}$ satisfying the conditions of the problem statement while considering:

$$U_k := \{u_k \mid \exists \ x_k \in X_k^\delta : u_k = -K_k x_k + d_k\} \subseteq U. \tag{16}$$

The control law (15) leads to the following closed-loop dynamics for (8):

$$\begin{aligned} x_{k+1} &= A_{z_k} x_k + B_{z_k} \kappa_k(x_k) + G_{z_k} v_k & \text{(17a)}\\ &= \underbrace{(A_{z_k} - B_{z_k} K_k)}_{=:A_{cl,k,z_k}} x_k + B_{z_k} d_k + G_{z_k} v_k. & \text{(17b)} \end{aligned}$$

Starting from the initial distribution of the random variable $x_0 \sim \mathcal{N}(q_{x,0}, Q_{x,0})$, the state distribution according to the dynamics (17b) follows from the linear transformation of ellipsoids to:

$$\begin{aligned} q_{x,k+1} &= A_{cl,k,z_k} q_{x,k} + B_{z_k} d_k + G_{z_k} q_v & \text{(18a)}\\ Q_{x,k+1} &= A_{cl,k,z_k} Q_{x,k} A_{cl,k,z_k}^T + G_{z_k} Q_v G_{z_k}^T. & \text{(18b)} \end{aligned}$$

With (18), the confidence ellipsoid X_{k+1}^δ is obtained from:

$$X_{k+1}^\delta = \varepsilon(q_{x,k+1}, \underbrace{Q_{x,k+1} c_x}_{=:Q_{x,k+1}^\delta}). \tag{19}$$

In [16], the following semi-definite program has been introduced, which is solved for any $k \in \{0, \ldots, N\}$ to provide the controller tuples (K_k, d_k) and thus X_{k+1}^δ:

$$\min_{S_{k+1}, K_k, d_k} \ \text{trace}\left(\begin{bmatrix} S_{k+1} & 0 & 0 \\ 0 & w_1 \|q_{x,k+1}\| & 0 \\ 0 & 0 & w_2 \|u_k\| \end{bmatrix} \right) \tag{20a}$$

subject to:

$$\begin{aligned} & q_{k+1}^T L q_{k+1} - \rho q_{x,k}^T L q_{x,k} \leq \alpha_k & \text{(20b)}\\ & q_{k+1} = A_{cl,k,z_k} q_{x,k} + B_{z_k} d_k + G_{z_k} q_v & \text{(20c)}\\ & \alpha_k \leq \max_{l \in \{1, \ldots, k\}} \omega^l \alpha_{k-l} & \text{(20d)}\\ & \begin{bmatrix} S_{k+1} & A_{cl,k,z_k} Q_{x,k} & G_{z_k} Q_v \\ Q_{x,k} A_{cl,k,z_k}^T & Q_{x,k} & 0 \\ Q_v G_{z_k}^T & 0 & Q_v \end{bmatrix} \geq 0 & \text{(20e)}\\ & \text{trace}(S_{k+1}) \leq \text{trace}(Q_{x,k}) & \text{(20f)}\\ & \begin{bmatrix} (b_i - r_{u,i} d_k) I_n & -r_{u,i} K_k (Q_{x,k}^\delta)^{-\frac{1}{2}} \\ -(Q_{x,k}^\delta)^{-\frac{1}{2}} K_k^T r_{u,i}^T & b_i - r_{u,i} d_k \end{bmatrix} \geq 0 & \text{(20g)}\\ & \forall i = \{1, \ldots, n_c\} & \text{(20h)} \end{aligned}$$

The cost function (20a) considers the size of the confidence ellipsoid X_{k+1}^δ, the Euclidean distance of the mean to the origin, and the input energy. The linear constraints (20b)-(20d) ensure a convergence of the mean value $q_{x,k+1}$ to the origin for increasing k. S_{k+1} in (20e) is an over-approximation of the covariance matrix $Q_{x,k+1}$, and (20f) enforces a reduction of all semi-axes of X_k^δ. The input constraint (16) is formulated by (20g). A detailed derivation of each constraint can be found in [16].

IV. Reachability analysis of PWAP Systems

The SDP as specified by (20) does not consider the partition of the continuous state space for the PWAP system, i.e., it does not take into account whether X_{k+1}^δ is completely contained in region $\Theta^{(i)}$. If containment applies ($X^\delta_{k+1} \cap \Theta^{(i)} = X^\delta_{k+1}$ for one $i \in Z$), the procedure continues for the next time step. If not, i.e. if X^δ_{k+1} intersects with two or more partition elements ($X^\delta_{k+1} \cap \Theta^{(i)} \neq X^\delta_{k+1}$), the following function is employed to determine all discrete states for which intersections of $\Theta^{(i)} \in \bar{\Theta}$ and X^δ_{k+1} exist:

$$Z_{int,k+1} := getIntReg(X^\delta_{k+1}, \bar{\Theta}) \subset Z \tag{21}$$

If the result is a singleton, i.e. for the cardinality $|Z_{int,k+1}|$ applies $|Z_{int,k+1}| > 1$, no further action is needed. Otherwise, two options are evaluated: (i.) to push X^δ_{k+1} into a single region $\Theta^{(i)}$, or (ii.) to use the multiple intersection to consider different branches of evolution for the following time steps. The first option, which is preferable in terms of computational effort is tried first. For this purpose, the initial SDP is modified to avoid intersection with the region boundary in time k: the SDP is solved again with an additional constraint to push X^δ_{k+1} into the desired direction. Fig. 1 shows this case on the left. To obtain one of the dotted ellipsoid (instead of X^δ_{k+1}), the additional constraint formulates that the distance between the center point $q_{x,k+1}$ and the boundary $\partial(\Theta^{(i)})$ is greater than the length of the semi-major axis of X^δ_k. If a hyperplane of $\partial(\Theta^{(i)})$ to which the distance has to be adjusted is parameterized by the normal vector ($r^{(i)}_j$ and the scalar $b^{(i)}_j$), this condition can be specified as:

$$\begin{aligned} d\left((r^{(i)}_j, b^{(i)}_j), q_{x,k+1}\right) &= r^{(i)}_j q_{x,k+1} - b^{(i)}_j \\ &\geq \sqrt{\max\{\Lambda(Q^\delta_{x,k})\}} \end{aligned} \tag{22}$$

The extended SDP, i.e. (20) with (22), has to be solved for every $i \in Z_{int,k+1} := getIntersecReg(X^\delta_{k+1}, \bar{\Theta})$, until a feasible solution is found. If multiple feasible solution exists for $i \in Z_{intsec,k+1}$, the best solution according to the cost function (20a) is chosen.

If the first option fails, i.e. none of the SDP problems lead to an ellipsoid fully contained in one region, the second option (ii.) is applied. The set X^δ_{k+1} remains as initially computed and is partitioned (right part of Fig. 1): one SDP problem is solved for each $i \in Z_{int,k+1}$, and the obtained control tuples are applied to X^δ_k, leading to different branches of the further evolution. When computing the successor sets X^δ_{k+2,γ_i} of X^δ_{k+1}, the computation only considers the respective share of X^δ_{k+1} which corresponds to the intersection with Θ^i. The share ϵ_i is determined by the probability of x_k being inside of $\Theta^{(i)} \cap X^\delta_{k+1}$. Computing this probability relies on solving a multi-dimensional integral of the Gaussian probability density function over $\Theta^{(i)}$:

$$Pr\left(x_{k+1} \in \Theta^{(i)}\right) = \int_{\zeta \in \Theta^{(i)}} \mathcal{N}(q_{x,k+1}, Q_{x,k+1}) d\zeta. \tag{23}$$

While hard to solve analytically, this integral can be approximated by using a combination of $n_{x,i}$ univariate distributions, which can be evaluated by the cumulative distribution function. The approximation used in [18], [15] for evaluating so-called chance constraints, is also suitable for the problem on hand. With (23), the share ϵ_i can by obtained from:

$$\epsilon_i = \frac{Pr(x_{k+1} \in \Theta^{(i)})}{\sum_{i \in Z_{int,k+1}} Pr(x_{k+1} \in \Theta^{(i)})}, \quad \sum_{i \in Z_{int,k+1}} \epsilon_i = 1. \tag{24}$$

Let a function to accomplish the computation according to (24) be named $getProbPart(X^\delta_k, \Theta^{(i)})$.

In order to keep track of the different branches of the confidence sets, a tree structure can be used. A node of the tree represents a tuple $\gamma = \left(Pre_\gamma, Suc_\gamma, X^\delta_{k+1}, Z_{int,k+1}, \epsilon_\gamma\right)$, in which Pre_γ refers to the predecessor node, Suc_γ denotes the set of successor nodes. X^δ_{k+1} is the confidence reachable set associated to the node, $Z_{int,k+1}$ is again the set of indices of the regions intersecting with X^δ_{k+1}, and ϵ_γ is the share of the confidence δ for X^δ_k, see (24). The edges of the tree establish the successor relation between two subsequent confidence sets (e.g. $X^\delta_{k+1} \rightarrow X^\delta_{k+2,\gamma_1}$ in Fig. 1. If branching occurs, as in the case of the figure that X^δ_{k+2,γ_1} and X^δ_{k+2,γ_2} stem from X^δ_{k+1}, the distribution $x_{k+1} \sim \mathcal{N}(q_{x,k+1}, Q_{x,k+1})$ is converted into two distributions $x_{k+2,\gamma_1} \sim \mathcal{N}(q_{x,k+2,\gamma_1}, Q_{x,k+2,\gamma_1})$ and $x_{k+2,\gamma_2} \sim \mathcal{N}(q_{x,k+2,\gamma_2}, Q_{x,k+2,\gamma_2})$. More precisely, two control tuples $(K_{k+1,\gamma_1}, d_{k+1,\gamma_1})$ and $(K_{k+1,\gamma_2}, d_{k+1,\gamma_2})$ are determined such that the probabilities for $k+2$ are distributed according to the shares ϵ_1 and ϵ_2. The fact, that X^δ_{k+2,γ_1} is more likely to occur, since it is more likely for x_k to be inside $\Theta^{(1)}$ than in $\Theta^{(2)}$, is considered by $\epsilon_1 = 0.6 > \epsilon_2 = 0.4$ in the figure.

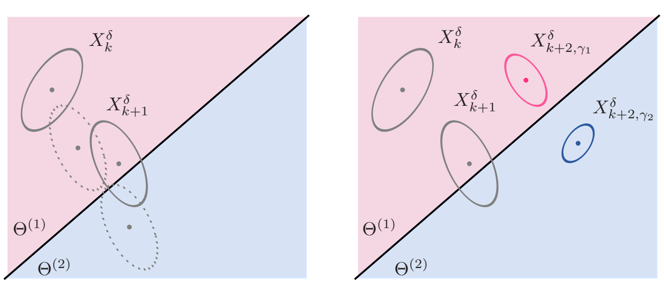

Fig. 1. Intersection of X^δ_{k+1} with more than one region $\Theta^{(i)}$: (i.) left: pushing X^δ_{k+1} into one region (dotted ellipsoids); (ii.) right: two branches arise from X^δ_{k+1} by solving an SDP problem each for the two dynamics assigned to $\Theta^{(1)}$ and $\Theta^{(2)}$ starting from X^δ_{k+1}.

V. Overall Synthesis Algorithm

The procedure to obtain the sequence of control laws $(K_{k,\gamma}, d_{k,\gamma})$ for $k \in \{1, \ldots, N\}$ to solve the problem 2.1 is now formulated as an algorithm. Let the set of nodes to be considered in time k be denoted by $\Gamma_k = \{\gamma_1, \ldots, \gamma_{n_{\gamma,k}}\}$. The Algorithm 5.1 steers the initial distribution $x_0 \sim \mathcal{N}(q_{x,0}, Q_{x,0})$ into the terminal region $\mathbb{T}$, and the computation terminates successfully with $k = N$, if the confidence ellipsoids $X^\delta_{N,\gamma} \in \mathcal{E}$ of all nodes $\gamma \in \Gamma_N$ are contained in the target set $\mathbb{T}$.

The main loop of the algorithm is executed until the terminal region $\mathbb{T}$ is not yet reached, and the confidence sets sufficiently approach $\mathbb{T}$ in the each step. The latter criterion,

Algorithm 5.1: **Probabilistic Control Algorithm**

given: (8) with $x_0 \sim \mathcal{N}(q_{x,0}, Q_{x,0})$, $v_k \sim \mathcal{N}(q_v, Q_v)$, $\bar{\Theta}$, and $U = \{u_k \mid Ru_k \leq b\}$; $\mathbb{T}$, δ, π_{min}, ω, ρ, and α_0
define: $k := 0$, $Z_{int,0} = getIntReg(X_0^\delta, \bar{\Theta})$,
$\pi_0 := \pi_{min}$, $\gamma_1 = (\emptyset, \emptyset, X_0^\delta, Z_{int,0}, 1)$, $\Gamma_0 := \{\gamma_1\}$
while $\exists\, \gamma \in \Gamma_k$ with $X_k^\delta \nsubseteq \mathbb{T}$ and $\pi_k \geq \pi_{min}$ **do**
  $\Gamma_{k+1} := \emptyset$
  for $\gamma_i \in \Gamma_k$ **do**
    for $p \in Z_{int,k}$ **do**
      solve the optimization problem (20) with $z_k = p$
$\star$ compute the distribution of $x_{k+1,p}$ from (17b)
      compute $X_{k+1,p}^\delta$ according to (19)
      $Z_{int,k+1} := getIntReg(X_{k+1,p}^\delta, \bar{\Theta})$
      if $|Z_{int,k+1}| > 1$ **do**
        "push" $X_{k+1,p}^\delta$ into one region by solving the SDP with the additional constraint (22)
        if a feasible solution exists **do** go to line $\star$
        else
          for $j \in Z_{int,k+1}$ **do**
            $\epsilon_j := getProbPart\left(X_{k+1,p}^\delta, \Theta^{(j)}\right) \cdot \epsilon(\gamma_i)$
            $\gamma_j := (\gamma_i, \emptyset, X_{k+1,p}^\delta, Z_{int,k+1}, \epsilon_{\gamma_j})$
            $Suc_{\gamma_i} := Suc_{\gamma_i} \cup \gamma_j$
            $\Gamma_{k+1} := \Gamma_{k+1} \cup \gamma_j$
          end
        end
      end
    end
  end
  compute π_{k+1} according to (25)
  $k := k + 1$
end while
return $(K_{k,\gamma}, d_{k,\gamma})$ for all $\gamma \in \Gamma_k$ and $0 \leq k \leq N-1$

which is included to avoid an unreasonably large number of iterations, can be modeled by:

$$\pi_{k+1} = || \min_{\gamma \in \Gamma_{k+1}} q_{x,k+1,\gamma} - \min_{\gamma \in \Gamma_k} q_{x,k,\gamma} || \geq \pi_{min} \qquad (25)$$

with a parameter $\pi_{min} \in \mathbb{R}$.

Lemma 5.1: Problem 2.1 is successfully solved, if Algorithm 5.1 terminates with $X_{N,\gamma}^\delta \subseteq \mathbb{T}, \forall \gamma \in \Gamma_N$. The solution provides a sequence of control laws (15) which steer any initial state $x_0 \in X_0^\delta$ with probability δ into the target set $\mathbb{T}$ in N steps. Furthermore, the input constraint $u_k \in U$ holds for all $0 < k < N$.

Sketch of Proof: For any $k \in \{0, 1, \ldots, N-1\}$ one of the following cases applies: (1.) X_k^δ is mapped by (18), (19) into $X_{k+1}^\delta \subseteq \Theta^{(i)}$ for one $\Theta^{(i)} \in \bar{\Theta}$; (2.) X_k^δ is mapped into $X_{k+1}^\delta \nsubseteq \Theta^{(i)}$ for any $\Theta^{(i)} \in \bar{\Theta}$, but the step of "pushing" leads to the same outcome as (1.); (3.) projecting X_k^δ forward in time leads to a set $Z_{int,k+1} \neq \emptyset$, pushing fails and branching occurs. For the cases (1.) and (2.), the solution of (20) by definition preserves the confidence δ, since the scaling of $X_{k+1,\gamma}^\delta = \varepsilon(q_{x,k+1,\gamma}, Q_{x,k+1,\gamma} c_x)$ by c_x adjust the size of the ellipsoid such that the share δ of the distribution $x_k \sim \mathcal{N}(q_{x,k}, Q_{x,k,})$ is transferred into X_{k+1}^δ (see also [16]). The same reasoning holds for all $k \in \{0, 1, \ldots, N-1\}$, such that $Pr(x_N \in X_N^\delta) = \delta$ follows from induction. For case (3.), the assignment for ϵ_j in the algorithm multiplies the probability of $x_k \in X_{k,\gamma_i}^\delta$ (i.e. ϵ_{γ_j}) with the share of intersection of $X_{k+1,p}^\delta$ with $\Theta^{(j)}$. According to (24), the union of the confidence ellipsoids reached from $X_{k+1,p}^\delta$ is again ϵ_{γ_j}. Thus, a branching step does not change the overall probability of including x_{k+1} in the confidence sets obtained from branching. With that the induction holds as for the cases (1.) and (2.). The satisfaction of the input constraints follows from the construction of (20). $\square$

In addition, attractivity with confidence δ and stability with confidence δ as defined in [16] follows under certain conditions from a successful termination of Algorithm 5.1.

VI. Numerical Example

To illustrate the principle of the method, assume that an automated vehicle (like a ship with autopilot) moves in a plane to reached a target region. Let the dynamics be approximately modeled by affine dynamics which differs in different regions of the plane. The corresponding PWAP system comprises three regions $\Theta^{(i)}$, two continuous states, and two continuous inputs. The regions can be seen from the bold lines in Fig 2. Let the initial distribution of the continuous state and the disturbances be given by:

$$x_0 \sim \mathcal{N}(q_{x,0}, Q_{x,0}) \text{ with } q_{x,0} = \begin{bmatrix} -10 \\ 50 \end{bmatrix}, \quad Q_{x,0} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

$$v_k \sim \mathcal{N}(0, Q_v) \text{ with } Q_v = \begin{bmatrix} 0.02 & 0.01 \\ 0.01 & 0.02 \end{bmatrix},$$

and the continuous dynamics by:

$$A_1 = \begin{bmatrix} 9.41 & 0.19 \\ -0.38 & 9.99 \end{bmatrix} 10^{-1}, \; B_1 = \begin{bmatrix} 1.98 & 0.02 \\ 3.96 & 2.00 \end{bmatrix} 10^{-1}$$

$$A_2 = \begin{bmatrix} 9.22 & 0.19 \\ -0.58 & 10.4 \end{bmatrix} 10^{-1}, \; B_2 = \begin{bmatrix} 1.96 & 0.02 \\ 4.02 & 2.04 \end{bmatrix} 10^{-1}$$

$$A_3 = \begin{bmatrix} 11.2 & -0.21 \\ 0.42 & 9.79 \end{bmatrix} 10^{-1}, \; B_3 = \begin{bmatrix} 2.12 & -0.04 \\ 0.04 & 3.96 \end{bmatrix} 10^{-1}$$

$$G_1 = G_2 = G_3 = \begin{bmatrix} 0.1 & 0.05 \\ 0.08 & 0.2 \end{bmatrix}.$$

The second and third subsystems have unstable state matrices. The inputs u_k are constrained according to: $-4 \leq u_{1,k} \leq 4$, $-8 \leq u_{2,k} \leq 4$. and the target set defined to: $\mathbb{T} = \varepsilon\left(0, \begin{bmatrix} 0.96 & 0.64 \\ 0.64 & 0.8 \end{bmatrix}\right)$. The algorithm 5.1 is parametrized by $\delta = 0.95$, $\pi_{min} = 0.01$, $\alpha_0 = 10^{-4}$, $\omega = 0.8$ and $\rho = 0.98$. The cost function is selected to $J = \text{trace}\left(\begin{bmatrix} S_{k+1} & 0 \\ 0 & w_1 \|q_{k+1}\| \end{bmatrix}\right)$ with $w_1 = 0.8$ (and $w_2 = 0$).

For the ship example mentioned in Sec. I, the PWAP model can be interpreted as different ship dynamics linearized in different areas of the sea, assuming that wind and water currents (as well as resulting model uncertainties) differ regionally. The control task is to transfer the vessel

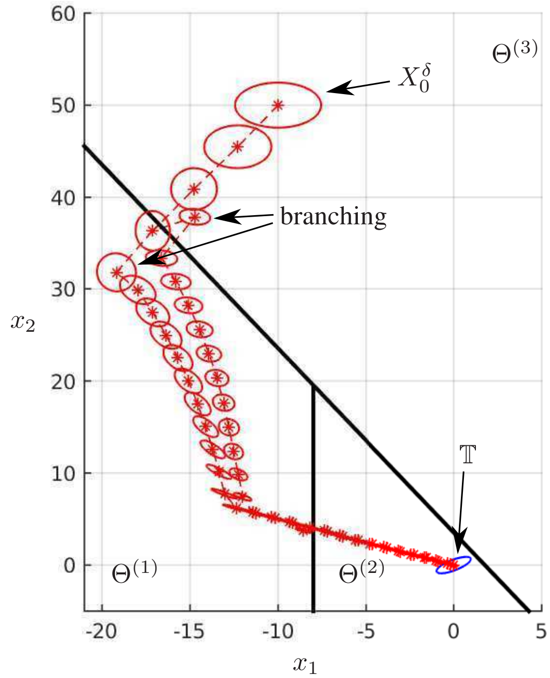

Fig. 2. Control result for the example: The initial confidence set with mean vector $[-10,\, 50]^T$ is steered to the origin in 30 iterations. Branching occurs after 3 steps with $\epsilon(\gamma_1) = 0.89$ and $\epsilon(\gamma_2) = 0.11$.

from an initial position, which might be randomly distributed around an expected value (due to imprecise localization) to a target region. The proposed Algorithm 5.1 is able to compute a control law while considering the different dynamics in each region and the stochastic disturbances.

The successful termination the synthesis algorithm is shown in Fig. 2, illustrating the confidence reachable sets X_k^δ. Three iterations after starting from X_0^δ the ellipsoid cannot be pushed into one region, thus X_3^δ intersects with $\Theta^{(3)}$ and $\Theta^{(1)}$, i.e. branching is required with shares $\epsilon(\gamma_1) = 0.89$ and $\epsilon(\gamma_2) = 0.11$. For the subsequent iterations, the SDP has to be solved for the two branches, while the evolution of the confidence sets converge to each other. This attractiveness follows from the Lyapunov condition (20b) within the SDP problems. The transition from $\Theta^{(1)}$ to $\Theta^{(2)}$ proceeds without branching. After $N = 30$ iterations, the confidence reachable sets of both branches are contained in the terminal set $\mathbb{T}$, and the algorithm terminates successfully.

The overall time of computation is 48 sec on a standard PC (Intel Core $i7-6700$ CPU, 16GB RAM, and Matlab $2016a$). The SDP problem is built with YALMIP and solved by MOSEK. So far, numerical examples up to a dimension of $n = 6$ have been successfully solved with Algorithm 5.1.

VII. Conclusion

This paper presents an algorithmic procedure to synthesize stabilizing control laws for PWAP systems. The procedure is based on probabilistic reachable set computation, and the proposed procedure formulates a semi-definite program in each time step to obtain a control law for set-to-set transitions of confidence reachable sets. The challenging case, where the confidence sets intersect with more than one partition is tackled by a branching procedure, which splits the confidence sets. To avoid the additional computational effort obtained for branching, a preceding step aims at pushing the confidence sets into one region. If this attempt fails, branching is unavoidable. The worst-case situation is that branching is needed in every time step k, leading to exponential growth of the computational effort. Thus, future research will explore possibilities of merging different branches using appropriate Lyapunov functions with the SDP problems.

Acknowledgements

Partial financial support by the European Commission through the project UnCoVerCPS (grant No. 643921) is gratefully acknowledged.

References

[1] L. Asselborn and O. Stursberg, "Probabilistic control of switched linear systems with chance constraints," pp. 2041–2047, 2016.

[2] E. D. Sontag, "Nonlinear regulation: The piecewise linear approach," *IEEE Trans. on Automatic Control,*, vol. 26, no. 2, pp. 346–358, 1981.

[3] S. V. Rakovic, E. C. Kerrigan, D. Q. Mayne, and J. Lygeros, "Reachability analysis of discrete-time systems with disturbances," *IEEE Transactions on Automatic Control*, vol. 51, no. 4, pp. 546–561, 2006.

[4] A. Bemporad, F. D. Torrisi, and M. Morari, "Optimization-based verification and stability characterization of piecewise affine and hybrid systems," in *Hybrid Systems: Comp. and Control*. Springer, 2000, pp. 45–58.

[5] O. Stursberg and B. Krogh, "Efficient representation and comp. of reachable sets for hybrid systems," in *Hybrid Systems: Comp. and Control*. Springer, 2003, pp. 482–497.

[6] A. Girard, "Reachability of uncertain linear systems using zonotopes," in *Hybrid Systems: Comp. and Control*. Springer, 2005, pp. 291–305.

[7] A. Girard, C. Le Guernic, and O. Maler, "Efficient comp. of reachable sets of linear time-invariant systems with inputs," in *Hybrid Systems: Comp. and Control*. Springer, 2006, pp. 257–271.

[8] C. Le Guernic and A. Girard, "Reachability analysis of linear systems using support functions," *Nonlinear Analysis: Hybrid Systems*, vol. 4, no. 2, pp. 250–262, 2010.

[9] L. Habets, P. J. Collins, and J. H. van Schuppen, "Reachability and control synthesis for piecewise-affine hybrid systems on simplices," *IEEE Trans. on Automatic Control*, vol. 51, no. 6, pp. 938–948, 2006.

[10] L. C. Habets and J. H. van Schuppen, "Control of piecewise-linear hybrid systems on simplices and rectangles," in *Hybrid Systems: Comp. and Control*. Springer LNCS, 2001, vol. 2034, pp. 261–274.

[11] E. C. Kerrigan and D. Q. Mayne, "Optimal control of constrained, piecewise affine systems with bounded disturbances," pp. 1552–1557, 2002.

[12] S. V. Rakovic, E. C. Kerrigan, and D. Q. Mayne, "Optimal control of constrained piecewise affine systems with state-and input-dependent disturbances," pp. MP8–01.25, 2004.

[13] X. D. Koutsoukos and P. J. Antsaklis, "Safety and reachability of piecewise linear hybrid dynamical systems based on discrete abstractions," *Discrete Event Dynamic Systems*, vol. 13, no. 3, pp. 203–243, 2003.

[14] L. Asselborn, D. Gross, and O. Stursberg, "Control of uncertain nonlinear systems using ellipsoidal reachability calculus," pp. 50–55, 2013.

[15] L. Asselborn and O. Stursberg, "Probabilistic control of uncertain linear systems using stochastic reachability," *8th IFAC Symp. on Robust Control Design*, vol. 48, no. 14, pp. 167–173, 2015.

[16] ——, "Robust control of uncertain switched linear systems based on stochastic reachability," pp. 310–316, 2015.

[17] W. J. Krzanowski and F. H. C. Marriott, *Multivariate Analysis: Distributions, Ordination and Inference*. Wiley Interscience, 1994.

[18] L. Blackmore and M. Ono, "Convex chance constrained predictive control without sampling," pp. 7–21, 2009.