

# Region of Attraction Estimation for a Perching Aircraft: A Lyapunov Method Exploiting Barrier Certificates

Elena Glassman, Alexis Lussier Desbiens, Mark Tobenkin, Mark Cutkosky, and Russ Tedrake

**Abstract**—Dynamic perching maneuvers for fixed-wing aircraft are becoming increasingly plausible due to recent progress in perching using ‘micro-spines’ mounted on tuned suspensions and, separately, on feedback motion planning techniques for post-stall maneuvering. In this paper, we bring these complementary techniques together by efficiently estimating the mechanical stability of the plane when it makes contact with a vertical surface; the resulting landing *funnel* can then be used in a feedback motion planning algorithm for the flight controller.

We consider a simplified model of the perching dynamics and report an extension of the region of attraction techniques, using sums-of-squares optimization, which combines polynomial approximations of barrier constraints with the traditional Lyapunov methods to achieve tight estimation of the true region of attraction for the model. We demonstrate the new method on a variety of design parameters for the perching system, suggesting a potential use as a mechanical system or controller design tool.

## I. INTRODUCTION

The work described here is aimed at enabling small UAVs to operate at the transition between air and surface contact, with walls, roofs and power lines. Such operation is of particular value for small planes, with wing-spans of 1 m or less, due to their severely limited mission life. They have lower lift/drag ratios than larger planes [1] and carry less energy aboard. Consequently, a state of the art plane such as the Aerovironment Black Widow has an endurance of 30 minutes [2]. The scaling laws that reduce flying time also favor frequent, even abrupt, landings due to the high specific strength of small components [3]. Indeed, most small UAVs today crash land. Similar scaling rules apply to biological fliers; not surprisingly, many small birds and other fliers perch frequently between short flights.

Recent work from the authors has addressed the problems of precise modeling and control for perching on wires [4, 5] and the design and analysis of a suspension that allows a wide range of touchdown states (pitch, pitch rate, horizontal and vertical velocities) for landing on rough walls, even with limited sensing [6]. In this paper, we aim to bring these complementary techniques together by analyzing and exploiting the mechanical design of the landing mechanism in order to allow for aggressive controllers and generally, more robust performance. To do so, one can use the feedback motion planning strategy called LQR-Trees [7], in which



Fig. 1. Perching sequence.

locally-valid flight controllers are analyzed to determine their *funnels*—the regions of state space in which, when applied, the controllers can be guaranteed to achieve the goal. These can be combined easily with the landing gear, assuming that we can estimate the *funnel*, or touchdown envelope for the landing gear.

Previous attempts to evaluate that touchdown envelope required a fine discretization of the state space (4D) and the simulation of each individual case, as described in [6]. This renders the optimization of the touchdown envelope prohibitive, as each change of the mechanical system requires multiple hours of simulation to fully determine its effect. As a result, the suspension is typically manually tuned using simple heuristics, and due to the limited search, it is hard to know if a better solution can be exploited. In this paper, we extend the theory required to estimate the region of attraction (ROA) in the presence of polynomial barrier certificates, with the specific goal of providing a tight estimate of the ROA of our perching model. These polynomial barrier functions include, but are not limited to, barriers between different states of the hybrid dynamical system formed by the plane making and possibly breaking contact with the wall. As will be illustrated in this paper, these techniques can efficiently estimate the region of attraction, without simulations, and be used as a design tool to evaluate and optimize a mechanical system performance.

## II. RELATED WORK

The use of computational models to analyze reliability and potential failure of designs has become ubiquitous in modern engineering. Simulation studies have proven to be a powerful tool both for analysis of abstract robotic models [8], and to guide the design of physical systems. As system complexity increases, exhaustive sample based simulation suffers from

E. Glassman, M. Tobenkin, and R. Tedrake are with the Computer Science and Artificial Intelligence Lab (CSAIL), MIT, Cambridge, MA 02139, USA {elg, mmt, russt}@mit.edu

A. Lussier Desbiens and M. Cutkosky are with the Biomimetics and Dextrous Manipulation Laboratory (BDML), Stanford University, Stanford, CA 94303, USA {alexisd, cutkosky}@stanford.edu

the curse of dimensionality and can take unmanageable time to accurately provide a measure of stability.

For computing regions of attraction of dynamical systems one family of alternatives requires solutions of non-linear partial differential equations (PDE). In this spirit, the approximate solution of Hamilton-Jacobi-Isaacs PDEs has been employed to analyze when hybrid robotic systems with disturbances can avoid certain “keep out” sets [9]. Other, discretization based methods have been employed to approximate continuous state dynamical systems, such as the “cell-to-cell” mapping techniques of [10], which improved brute-force discretizations of a state space. More recent work analyzing hybrid systems has exploited connections to tools from automata theory [11].

The techniques presented here build directly upon recently developed tools for estimating regions of attraction via convex optimization. These techniques replace solving partial differential equations with partial differential *inequalities*. For systems involving polynomial functions it was noted that such problems can be addressed conservatively via convex optimization [12]. Our approach particularly augments coordinate ascent based region of attraction optimization techniques presented in [13] to include “barrier” functions representing undesired transitions or collisions for a perching robot. In [14] such barrier functions were optimized for stochastic dynamical systems via coordinate ascent when a “keep out” region could be specified *a priori*.

### III. A SIMPLE POST-TOUCHDOWN MODEL

In order to calculate the true ROA of the airplane during landing, a simplified version from the model presented in [6] will be used. In that work, it was shown experimentally that a model similar to what will be described in this section correctly approximates the forces experienced during landing and the ROA of the airplane. The model presented here has fewer degrees of freedom (DOF) than the model derived in [6] but still exhibits rich dynamics similar to those experienced by the more complex model. The reduced number of DOFs allows for an easier visual verification of the approximated touchdown envelope, a useful feature during the development of the tools described in this paper.

This simple model of the post-touchdown configuration models the airplane as a rigid body subject to gravity, attached to the wall at the foot with a massless leg and attached at the tail with a frictionless slider joint, as illustrated on Figure 2. As on the real airplane, the suspension is passive, incorporating only a linear spring and damper. As the velocity at touchdown is usually small ( $< 3$  m/s), the aerodynamic forces are small in comparison to the other forces and thus neglected.

In order to simplify the notation, three right-handed reference frames are introduced: the wall frame  $\mathcal{W}$  with the unit vector  $\hat{x}_{\mathcal{W}}$  oriented toward the wall and  $\hat{y}_{\mathcal{W}}$  upward along the surface; the airplane frame  $\mathcal{A}$  is rotated by  $q_A$  from  $\mathcal{W}$ , around  $\hat{z}_{\mathcal{W}}$ ; the femur frame  $\mathcal{F}$  is rotated by  $-q_H$  from  $\mathcal{A}$ , around  $\hat{z}_{\mathcal{W}}$ . The following paragraphs describe the procedure used to derive the equations governing the model

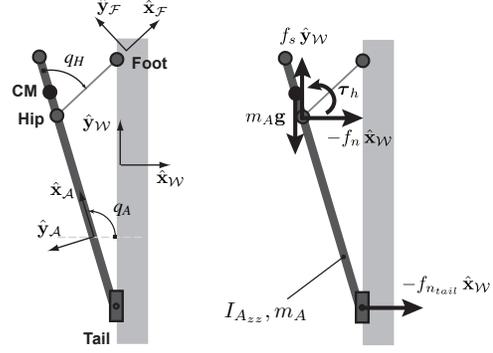


Fig. 2. Simplified model of the airplane. The plane is modeled as a rigid bar and the suspension as a massless link connected through linear spring and damper at the hip joint. Reference frames and variables are illustrated on the left, while forces and torques applied on the airplane are illustrated on the right.

and the constraints that should be respected for successful landing. Due to their excessive length, the explicit equations are not included in this paper. However, their derivation with a software like Motion Genesis<sup>TM</sup> [15] is trivial once the identifiers (see Table I), the kinematics, the forces and the constraints are defined.

TABLE I  
IDENTIFIERS FOR THE SIMPLE AIRPLANE MODEL

variables:		
$q_A$	-	airplane pitch: from $\hat{x}_{\mathcal{W}}$ to $\hat{x}_{\mathcal{A}}$ , along $\hat{z}_{\mathcal{W}}$
$q_H$	-	hip angle: from $\hat{x}_{\mathcal{A}}$ to $\hat{x}_{\mathcal{F}}$ , along $\hat{z}_{\mathcal{A}}$
$f_n$	-	adhesion force acting on the foot, along $-\hat{x}_{\mathcal{W}}$
$f_s$	-	shear force acting on the foot, along $\hat{y}_{\mathcal{W}}$
$f_{n_{tail}}$	-	normal force acting on the tail, along $-\hat{x}_{\mathcal{W}}$
nominal parameters:		
$m_A$	0.4 kg	mass of the airplane
$I_{A_{zz}}$	0.0164 kgm <sup>2</sup>	moment of inertia of the airplane around $\hat{z}_{\mathcal{A}}$
$b_h$	0.0012 Nms/°	damping coefficient at the hip joint
$k_h$	0.0041 Nm/°	spring stiffness at the hip joint
constants:		
$l_f$	0.15 m	length of the leg, hip to foot, along $\hat{x}_{\mathcal{F}}$
$l_h$	-0.03 m	dist. from plane COM to hip, along $\hat{x}_{\mathcal{A}}$
$l_t$	0.57 m	dist. from plane COM to tail, along $-\hat{x}_{\mathcal{A}}$
$g$	9.81 m/s <sup>2</sup>	gravitational acceleration
$q_{h_0}$	45°	natural hip angle
$\alpha$	1	adhesion limit for asphalt roofing shingle
$\mu$	0.3	coefficient of static friction

The first step to derive the equations needed is to define the acceleration of the center of mass. To do so, the position of the center of mass ( $\mathbf{r}^{ACM/W_0} = l_h \hat{x}_{\mathcal{A}} - l_f \hat{x}_{\mathcal{F}}$ ), can be differentiated twice with respect to time and in reference frame  $\mathcal{W}$ :

$$\begin{aligned} \mathcal{W}_{\mathbf{a}}^{ACM} &= l_f (\dot{q}_H - \dot{q}_A)^2 \hat{x}_{\mathcal{F}} + l_h \ddot{q}_A \hat{y}_{\mathcal{A}} + \dots \\ &\quad l_f (\ddot{q}_H - \ddot{q}_A) \hat{y}_{\mathcal{F}} - l_h \dot{q}_A^2 \hat{x}_{\mathcal{A}} \end{aligned} \quad (1)$$

The forces acting on the airplane are the gravity ( $\mathbf{g} = -m_A g \hat{y}_{\mathcal{W}}$ ) acting at the center of mass, and the contact forces transmitted from the foot to the hip joint by the massless leg ( $\mathbf{f}_{h,contact} = -f_n \hat{x}_{\mathcal{W}} + f_s \hat{y}_{\mathcal{W}}$ ). Due to the spring and damper located at the hip, the hip torque is

$\boldsymbol{\tau}_h = -(k_h(q_h - q_{h0}) + b_h\dot{q}_h)\hat{\mathbf{z}}_{\mathcal{W}}$ . At the frictionless slider joint located at the tail, the force is  $\mathbf{f}_{tail} = -f_{ntail}\hat{\mathbf{x}}_{\mathcal{W}}$ . Knowing these, it is possible to write the equations of motion by equating the sum of the forces to the effective force and the sum of the moments of force around the hip point to the moment of effective force around the same point [16]:

$$\mathbf{f}_{h,contact} + \mathbf{g} + \mathbf{f}_{tail} = m_A {}^{\mathcal{W}}\mathbf{a}^{ACM} \quad (2)$$

$$\begin{aligned} \boldsymbol{\tau}_h + l_h \hat{\mathbf{x}}_A \times \mathbf{g} - l_t \hat{\mathbf{x}}_A \times \mathbf{f}_{tail} = \dots \\ \mathbf{I}^{A/Ahip} \cdot {}^{\mathcal{W}}\boldsymbol{\alpha}^A + m_A \mathbf{r}^{ACM/Ahip} \times {}^{\mathcal{W}}\mathbf{a}^{Ahip} \end{aligned} \quad (3)$$

where  $\mathbf{I}^{A/Ahip}$  is the inertia dyadic of the airplane around the hip point,  ${}^{\mathcal{W}}\boldsymbol{\alpha}^A$  is the angular acceleration of the airplane in  $\mathcal{W}$  and  $\mathbf{r}^{ACM/Ahip}$  is the position vector of the airplane center of mass with respect to the hip joint. From these vectorial equations, it is possible to obtain 3 scalar equations by dotting equation 2 with  $\hat{\mathbf{x}}_{\mathcal{W}}$  and  $\hat{\mathbf{y}}_{\mathcal{W}}$ , and equation 3 with  $\hat{\mathbf{z}}_{\mathcal{W}}$ . A fourth equation can be written by dotting the moment balance around the massless leg with  $\hat{\mathbf{z}}_{\mathcal{W}}$ :

$$-(k_h(q_h - q_{h0}) + b_h\dot{q}_h) = (l_f \hat{\mathbf{x}}_{\mathcal{F}} \times \mathbf{f}_{h,contact}) \cdot \hat{\mathbf{z}}_{\mathcal{W}} \quad (4)$$

Finally, an additional kinematic constraint is used to enforce the slider joint at the tail:

$$\frac{d^2}{dt^2} \left( \mathbf{r}^{Atail/W_0} \cdot \hat{\mathbf{x}}_{\mathcal{W}} = 0 \right) \quad (5)$$

These 5 equations are then used to solve for  $\ddot{q}_A$ ,  $\ddot{q}_H$ ,  $f_n$ ,  $f_s$  and  $f_{ntail}$ . Note that in this case,  $q_H$  is driven by  $q_A$ , due to the preceding kinematic constraint. The angular velocity  $\dot{q}_H$  could also be solved for directly by setting the velocity of the tail along  $\hat{\mathbf{x}}_{\mathcal{W}}$  to be zero.

1) *Constraints:* During landing, the system must satisfy various constraints to stay attached to the wall and remain in the desired configuration. For the spines to stay attached, the forces must stay within the green safe zone in Figure 3. This means that the forces must respect the friction limit and the adhesion limit, which are functions of the asperity shape and material properties. The force at the foot must also not exceed the maximum force that the asperities can sustain. These conditions are listed in Table II; more details about spine interaction can be found in [6, 17].

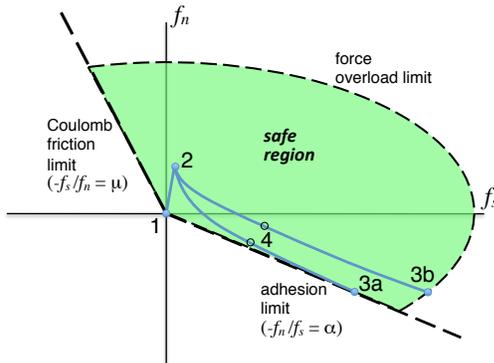


Fig. 3. Force space representation of the constraints that must be respected for the spines to stay attached to the wall surface.

In addition, the tail must remain on the wall by maintaining  $f_{ntail} \geq 0$ . It is also desirable to prevent the hip/nose from touching, by maintaining  $q_A \geq 90^\circ$ , as their contact with the wall means that the suspension failed to absorb the landing forces.

TABLE II  
POST-TOUCHDOWN CONSTRAINTS

Description	Constraint	Active when
Upward sliding	$-f_s \leq \mu f_n$	$f_s < 0, f_n > 0$
Spine adhesion	$-f_n \leq \alpha f_s$	$f_s > 0, f_n < 0$
Max. force	$\sqrt{(f_n/c)^2 + f_s^2} \leq f_{max}$	always
Tail rebound	$f_{ntail} \geq 0$	always
Nose hitting	$q_A \geq 90^\circ$	$\dot{q}_A \leq 0$

All these constraints are summarized in Table II and the barriers that they create on the dynamics of the system are illustrated in the left side of Figure 5.

#### A. Estimated ROA using previous techniques

The full ROA is illustrated in Figure 5 for two different airframes. The first design evaluated is the baseline airframe (400g) defined by the parameters of Table I, while the second design corresponds to a lighter version (200g). By sampling the state space, it is possible to instantaneously evaluate the constraints. The remaining points satisfying the constraints can then be simulated forward in time to determine if they will lead to failure. In both cases, it was found that the ROA is limited by the spine adhesion, tail force and the nose hitting constraints. In these specific cases, upward sliding, maximum force and tail rebound are not contributing to the shape of these ROAs.

In this simple 1DOF problem, it is straight forward to graph the constraints and the trajectories that are defining the ROA on a phase diagram and visually evaluate the envelope. This approach would not work in higher dimension and the discretization of the state space followed by numerous simulations would be required, as done in [6].

## IV. VERIFICATION WITHIN BARRIERS

In this section we describe how to extend previous techniques for computing ROAs of dynamical systems to handle the kinds of constraints presented above. We will deal with systems described by an ordinary differential equation:

$$\dot{x}(t) = f(x(t)), \quad (6)$$

representing autonomous (possibly closed loop) dynamics with state vector  $x(t) \in \mathbb{R}^n$ . We also are given a failure region,  $\mathcal{F} \subset \mathbb{R}^n$ , described by inequalities such as those in in Table II, and an exponentially stable equilibrium  $x_0 \in \mathbb{R}^n$  (i.e.  $f(x_0) = 0$ ). Our goal is to approximate the set of initial conditions for which solutions of (6) approach the equilibrium without ever passing through the failure region,  $\mathcal{F}$ .

### A. Lyapunov and Barrier Functions

We will find inner-approximations of this set via a combination of Lyapunov and barrier functions. Throughout this section we denote by  $V : \mathbb{R}^n \mapsto \mathbb{R}$  a positive, differentiable function such that  $V(x_0) = 0$ , and  $V(x) > 0$  when  $x \neq x_0$ . Lyapunov based approaches to ROA estimation generally rely on statements of the following sort [18]. If for some positive  $\epsilon$ :

$$V(x) \leq 1 \implies \frac{\partial}{\partial x} V(x) f(x) \leq -\epsilon V(x)$$

for all  $x \in \mathbb{R}^N$ , then the 1-sublevel set of  $V$ :

$$\Omega = \{x \mid V(x) \leq 1\}$$

belongs to the region of attraction and solutions which begin in this set never leave (i.e. the set is *positively invariant*). A naive initial approach to extending this analysis would simply be to require  $\Omega$  to be disjoint from the failure region  $\mathcal{F}$  (as solutions will never leave  $\Omega$ ). We found this approach to be very conservative, as our examples will bear out, and in the remainder of this section we develop our alternative.

We attempt to find a region defined by a family of smooth “barrier functions”  $B_i : \mathbb{R}^n \mapsto \mathbb{R}$  which excludes all the failure region. We will refer to the points where  $B_i(x) = 0$  as “barriers”. We design the set of  $m$  functions  $\{B_i\}_{i \in \mathcal{I}}$  (with  $\mathcal{I} = \{1, \dots, m\}$ ) so that the set:

$$\mathcal{S} = \{x \mid B_i(x) \geq 0, \quad \forall i \in \mathcal{I}\} \quad (7)$$

does not intersect the failure region. This allows us to form a new differential constraint to define a new inner approximation of the ROA. We instead require a set of  $m+1$  conditions to hold for all  $x \in \mathbb{R}^n$ :

$$x \in \Omega \cap \mathcal{S} \implies \dot{V}(x) \leq -\epsilon V(x), \quad (8)$$

$$x \in \Omega \cap \mathcal{S}, \text{ and } B_i(x) = 0 \implies \dot{B}_i(x) > 0 \quad \forall i \in \mathcal{I}. \quad (9)$$

The corresponding result we state without proof, which is that under these two conditions the set:

$$\Omega \cap \mathcal{S} = \{x \mid V(x) \leq 1, B_i(x) \geq 0, \forall i \in \mathcal{I}\} \quad (10)$$

belongs to the region of attraction and is again positively invariant.

In words, these conditions examine the intersection of the “safe set”  $\mathcal{S}$  defined by the barrier functions and the 1-sublevel set of  $V$  as the new inner approximation of the ROA. Whenever the boundary of this set includes part of a barrier (i.e. when  $x \in \Omega \cap \mathcal{S}$  and some  $B_i(x) = 0$ ) we require that the vector field flow *into the approximate ROA* via an additional differential constraint, as Figure 4 illustrates.

### B. Computational Overview

Our approach will be to iteratively improve an estimate of the ROA defined by a function  $V(x)$  as described above. We will use sum-of-squares (SOS) programming to verify the non-negativity conditions above, which requires the involved functions to be polynomials (see [12]). This is not particularly restricting for  $V(x)$ , and we describe out techniques

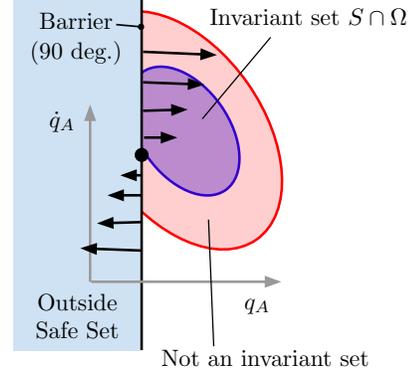


Fig. 4. An illustration of analysis with stability analysis and barriers. A vertical barrier function  $B_1(x)$  is negative for  $q_A$  is less than 90 degrees and positive otherwise representing the plane’s nose colliding with the wall. This defines a safe set  $\mathcal{S}$  to the right of the barrier where  $q_A$  is equal to 90 degrees. A positively invariant region in purple is given by the intersection of the safe set and the smaller ellipsoidal sub-level set of a Lyapunov function  $V$  (not fully pictured). The intersection with a larger ellipsoidal sub-level set would not be positively invariant, as further growth would include points on the boundary of the barrier where the flow of the dynamics (black arrows) would exit the safe set  $\mathcal{S}$ .

for constructing conservative polynomial barrier functions  $B_i(x)$  in Section V. This computation, combined with a local polynomial approximation of the dynamics  $f(x)$ , occurs once. To test various constraints on algebraic and semi-algebraic sets (i.e. sets defined by intersections of polynomial equalities and inequalities respectively) we make frequent use of the polynomial S-Procedure [19]. This technique reduces conservative verification of conditions such as (8) and (9) into a semidefinite program. Reducing the conservatism of the technique requires optimization over the coefficients of polynomial *multipliers*, which will be referenced later. These multipliers transform constrained positivity tests such as (8) and (9) into unconstrained tests in analogy to Lagrange multipliers in optimization.

In this work we’ll describe candidate  $V(x)$  as either being quadratic or quartic polynomials in the state  $x$ . The linear theory guarantees the existences of a locally valid quadratic Lyapunov function  $V_0(x) = (x - x_0)' P (x - x_0)$  where  $P$  is a symmetric, positive definite,  $n \times n$  matrix. This  $P$  is derived by solving a Lyapunov equation  $PA + A'P = -Q$  for another positive definite  $n \times n$  matrix  $Q$ , where  $A$  is the Jacobian of  $f(x)$  at the equilibrium. We begin our process by finding an appropriate scaling of  $P$  such that  $V_0(x) \leq 1$  satisfies our differential constraints (see Algorithm 1). This forms the initial Lyapunov candidate for our bilinear alternation (also called coordinate descent) approach.

Inspired by the choice of the volume of the region of attraction as a measure of stability, it is natural to attempt to maximize the volume of  $\Omega \cap \mathcal{S}$ . Unfortunately, even when  $\Omega$  is ellipsoidal this objective is generally non-convex. Further calculating the volume of a general semi-algebraic is computationally difficult. We follow the scheme proposed in Topcu et al. [13]. We produce a certificate that  $\Omega$  contains an ellipsoid of fixed orientation and take the maximal radius

of such ellipsoids to be the measure of the size of  $\Omega$ . We describe a general ellipsoid  $\mathcal{E}$  by the equations:

$$\mathcal{E} = \{x \in \mathbb{R}^n \mid x' S x + 2c'x + b \leq 0\}, \quad (11)$$

where  $S$  is a fixed positive definite matrix, and require:

$$x \in \mathcal{E} \implies V(x) \leq 1. \quad (12)$$

While we are interested in the volume of  $\Omega \cap \mathcal{S}$ , we have found allowing the contained ellipse to grow within the region  $\Omega$  allows for less conservative solutions.

### C. Algorithm

We now present the components of our algorithm which iteratively improve the ROA estimate through successive computations of  $V(x)$  and polynomial multipliers involved in the S-procedure which certify (9),(8),(12). These iterations improve the estimate in terms of the radius of the contained ellipse (11).

The Algorithm 1 provides a method of determining an initial *quadratic* function  $V(x) = (x - x_0)' P (x - x_0)$ , where  $P = P' \in \mathbb{R}^{n \times n}$  is a positive definite matrix. As a step of the algorithm, we solve the following optimization in  $\rho$ :

$$\text{maximize}_{\rho} \quad \rho \quad (13)$$

$$\text{subject to} \quad \|x - x_0\|^2 (V_0(x) - \rho) = 0 \implies \dot{V}_0(x) \leq -\epsilon V_0(x), \\ B_i(x) = 0 \implies V_0(x) > \rho, \quad \forall i \in \mathcal{I}.$$

Using the polynomial S-procedure the above is an optimization in the coefficients of a number of polynomial multipliers linear in the number of barriers and the single coefficient  $\rho$  (as  $V_0(x)$  is fixed). If we choose *fixed* polynomial multipliers for the terms involving  $\rho$  this is further a convex optimization. For the choice of  $V_0(x) = (x - x_0)' P_0 (x - x_0)$ , where  $P_0$  is the solution of the Lyapunov equation given in Algorithm 1, linear systems theory guarantees  $\dot{V}_0(x) < -\epsilon V_0(x)$  in a neighborhood of 0. The above program ensures that, except at  $x = x_0$ , if  $V_0(x) \leq \rho$  then  $\dot{V}_0(x) + \epsilon V_0(x) \neq 0$ , thus  $\dot{V}_0(x)$  must still be less than  $-\epsilon V_0(x)$ . Further we require that  $V_0(x) \leq \rho$  implies  $B_i(x) > 0$ . This guarantees that the set  $\Omega$  with  $V(x) \equiv V_0(x)$  is entirely contained in the safe set  $\mathcal{S}$ .

---

#### Algorithm 1 Find Initial Quadratic Lyapunov Function.

---

- 1: **procedure** INITIALQUADRATIC( $x_0, f, B$ )
  - 2:  $A \leftarrow \frac{\partial}{\partial x} f(x_0)$ .
  - 3:  $P_0 \leftarrow$  solution of  $A' P_0 + P_0 A = -I$ .
  - 4:  $\epsilon \leftarrow$  positive value less than smallest eigenvalue of  $P$ .
  - 5:  $V_0 \leftarrow (x - x_0)' P_0 (x - x_0)$ .
  - 6:  $\rho \leftarrow$  solution to optimization problem (13).
  - 7:  $P \leftarrow P_0 / \rho$
  - 8: **return**  $P$
  - 9: **end procedure**
- 

We now present the iterative technique which we apply for finding less conservative ROA estimates which satisfy (9),(8),(12). The initialization step provides us both with an

initial Lyapunov candidate  $V(x) = (x - x_0)' P (x - x_0)$  with  $P = \frac{1}{\rho} P_0$  and a contained ellipse:

$$\mathcal{E}_0 = \{x \in \mathbb{R}^n \mid x' S x + 2c_0'x + b_0 \leq 0\}, \quad (14)$$

given by  $S = P$ ,  $c_0 = -Sx_0$  and  $b_0 = x_0' S x_0 - r$  for any  $r < 1$ , as  $V$  is quadratic. Algorithm 2 describes the overall procedure.

---

**Algorithm 2** Given a polynomial differential equation  $f : \mathbb{R}^n \mapsto \mathbb{R}^n$  and set of barrier functions  $B = \{B_i\}_{i \in \mathcal{I}}$  optimize  $V$ .

---

- 1: **procedure** OPTIMIZEOMEGA( $x_0, f, B$ )
  - 2:  $P \leftarrow$  initialQuadratic( $x_0, f, B$ ).
  - 3:  $V \leftarrow (x - x_0)' P (x - x_0)$ .
  - 4:  $S \leftarrow P, c \leftarrow -P x_0, b \leftarrow x_0' P x_0 - r$ .
  - 5: **while** Stopping criterion not met. **do**
  - 6:  $(V, c, b) \leftarrow$  growQuartic( $x_0, f, B, V, S, c, b$ )
  - 7: **end while**
  - 8: **return**  $V$
  - 9: **end procedure**
  - 10: **procedure** GROWQUARTIC( $x_0, f, B, V_0, S, c_0, b_0$ )
  - 11:  $s \leftarrow$  multiplier polynomials verifying the conditions of (15) for  $(V, c, b) \equiv (V_0, c_0, b_0)$ .
  - 12:  $(V, c, b) \leftarrow$  solution of optimization problem (15), using multiplier polynomials  $s$ .
  - 13: **return**  $(V, c, b)$
  - 14: **end procedure**
- 

Lines 11 and 12 of Algorithm 2 involve solving two convex optimizations related to the program (15) below. The first optimization searches over the coefficients of the polynomial multipliers associated with applying the S-procedure to the family of polynomial constraints in (15). Here the number of multipliers grows quadratically with the number of barriers.

$$\text{maximize}_{V, b, c} \quad -b \quad (15)$$

$$\text{subject to} \quad V(x) \geq \epsilon_1 \|x - x_0\|^2, \\ x \in \Omega \cap \mathcal{S} \implies \dot{V}(x) \leq -\epsilon_2 V(x), \\ x \in \Omega \cap \mathcal{S}, B_i(x) = 0 \implies \dot{B}_i(x) \geq 0, \forall i \in \mathcal{I} \\ x' S x + 2c'x + b \leq 0 \implies V(x) \leq 1.$$

The second program minimizes  $b$  through the choice of  $b, c$  and  $V$ . In our examples we examine choosing  $V$  to be a quartic polynomial. While  $b$  is not technically the radius of the contained ellipse it is nonetheless monotonic in this radius. The choice of  $c \neq 0$  corresponds to examining ellipses whose center is not at  $x = 0$ . In our examples this has proven to be an important, albeit incremental, improvement on the original method proposed in [13]. We generally stop either after a maximum number of iterations or after the percent growth of the contained ellipse between iterations, measured by radius, is sufficiently slow.

## V. POLYNOMIAL APPROXIMATION USING SUPPORT VECTOR MACHINES

In order to solve the optimization problems posed in the previous section, it is necessary to find polynomial approximations of the dynamics,  $f(x)$ , and any of the barrier

functions,  $\{B_i(x)\}_{i \in \mathcal{I}}$ , which are non-polynomial. If one approximates a barrier  $B_i(\cdot)$  by a polynomial function  $\hat{B}_i(\cdot)$  it is possible to maintain conservatism so long as  $\hat{B}_i(x)$  is negative whenever  $B_i(x)$  is negative.

The dynamics of the plane-wall system about its perched fixed point are nonlinear. For the results in this paper, we chose a third-order Taylor expansion of the dynamics about that fixed point as the polynomial approximation to  $f(x)$ .

Several barrier functions in the plane-wall system which define surfaces separating one hybrid mode from another are also nonlinear. We chose to sample points on both sides of these barriers and then use the soft-margin Support Vector Machine (SVM) binary classifier algorithm with a polynomial kernel, as implemented by [20], to find polynomial approximations. Due to the structure of the cost function used in the optimization step of the SVM algorithm, the decision boundary will have an associated margin outside of which a misclassified point is costly but permitted by the existence of slack variables. While a clean option for ensuring conservatism may be modifying the slack variables themselves, the current solution for generating polynomial barriers that are conservative with respect to the samples is to tune the polynomial degree and cost function parameters such that no samples from the “unsafe” side of the nonlinear boundary to fall outside the margin on the “safe” side of the polynomial SVM class boundary, breaking the problem into smaller, simpler classification problems if necessary. Then the margin, which is the 1-level set of the decision barrier function returned by the SVM, is a conservative polynomial barrier approximation.

## VI. RESULTS

Using the approximated dynamics and barriers, and the method described in this paper, it is possible to estimate the ROA of the simple perching model described previously. The estimated ROA of two different airframes, the original system as well as a lighter version, are illustrated on the right side of Figure 5 and can be compared with the real ROA on the left side of the same figure.

Before discussing the estimated area, a few important observations should be made. First, one can observe that the approximated polynomial dynamics are generally well fitting, particularly in close proximity to the fixed point of the system as expected from a Taylor Expansion. The fit deteriorates around the strong non-linearity close to  $q_A = 104^\circ$ , as the legs fully straighten, but this region is not relevant as it is behind the barriers and represents states unreachable by the mechanical system. Second, the combination of linear and SVM approximated barriers provides a close estimate of the barriers acting on this system.

We computed estimates of the ROA using Algorithm 2. The SOS programs were processed into semidefinite programs using YALMIP [21] and solved using SeDuMi [22]. As expected, the algorithm allows the quartic to jump over the sections of the barriers where  $\hat{B}_i > 0$  and the quartic usually grows all the way to the point where  $\hat{B}_i$  becomes negative (**A** on Figure 5). Furthermore, the formulation

allows the quartic to grow unrestricted by the constraints once it has crossed one constraint. This behavior is shown by point **B** on Figure 5.

All of these factors are important to favor the growth of the estimated ROA. In both cases analyzed here, the estimated ROA area represents 78% of the ROA calculated from the real system. For a 400g platform, the estimated region has an area of  $894 \text{ deg}^2/\text{s}$  compared to  $1153 \text{ deg}^2/\text{s}$ . As the mass of the airplane is reduced to 200g, the real ROA increases in size to  $2211 \text{ deg}^2/\text{s}$ , and so does the area estimated by the quartic which reaches  $1735 \text{ deg}^2/\text{s}$ . The new method proposed in this paper produces significantly better ROA estimations than a method that would use a fixed ellipsoid limited in growth by the barriers, as illustrated by the blue dotted ellipsoid in Figure 5. For the two variations of the system presented in this paper, both the real ROA and the estimated polynomial ROA are suggesting that a lower mass is desirable to favor higher speed landings, which is consistent with our experience on the real airplane and suspension.

## VII. CONCLUSION

In the future, this approach could be automated and repeated for multiple parameters (e.g., joint stiffness and damping, leg length, inertia) and a design that leads to the largest ROA could be found. This has an important implication for the design of mechanical systems as it allows the designer to optimize the design for robustness to various operating conditions rather than performance for a single *typical* case, without recourse to numerous simulations.

More generally, our interest is not only in creating the largest *touchdown ROA* for the suspension, but the largest *flying ROA* that will connect to the *touchdown ROA*. This will allow the simultaneous evaluation of parameters like the mass and inertia of the airplane that have an influence during both the flight and touchdown phase. This will require us to deal with the other transitions experienced during the landing phase (e.g., sliding up, foot only touchdown) by using hybrid models.

There are natural extensions to the optimization tools presented here which could further improve our analysis. First, sums-of-squares optimization can be applied directly to the mixed trigonometric (in positions) and polynomial (in velocities) dynamics of the model, without requiring polynomial approximation; these optimization tools are less mature, but are progressing quickly. Second, if the model parameters (e.g. friction of the climbing surface) are unknown, it is natural to incorporate a (conservative) notion of robustness into this verification by requiring that the Lyapunov and barrier conditions are met by all possible vector fields given the uncertain system[23]. This technique can be used to capture real uncertainty about the perching environment, or known limitations in the simple models.

Finally, the tools described here should be applied to the full model of the perching airplane described in [6] and the predicted ROA evaluated on the real hardware. The approach remains the same as described here, but in higher dimension.

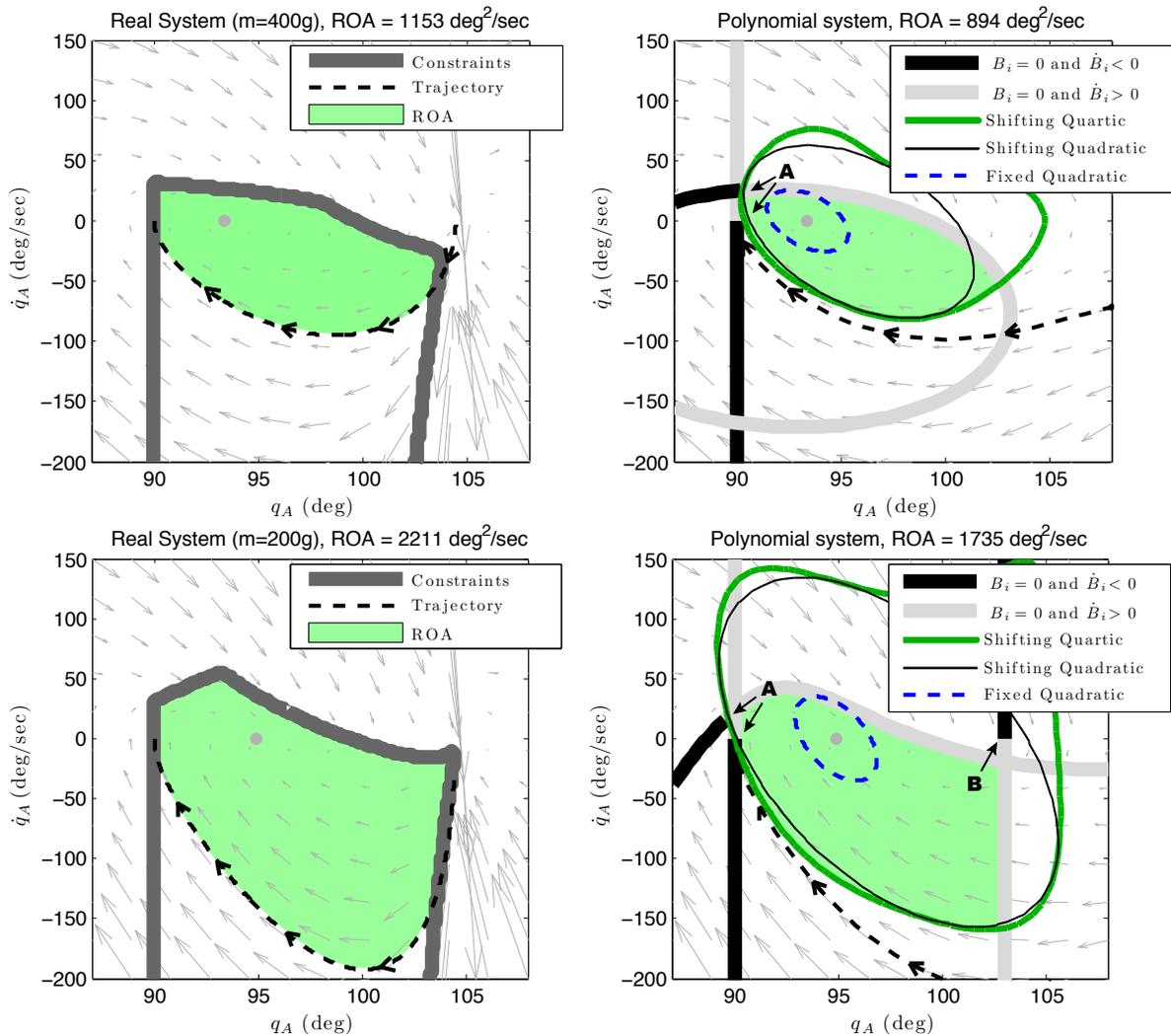


Fig. 5. Comparison of real and estimated ROA for two different cases:  $m_A = 0.4\text{kg}$  on top row, and  $m_A = 0.2\text{kg}$  on bottom row. The graphs on the left illustrate the real dynamics, aggregated constraints and corresponding ROA as the green shaded region. The real ROA is limited by the force on the tail, the  $f_n/f_s$  ratio and the fact that the nose shouldn't touch the wall at  $q_A = 90^\circ$ . The lower limit of the ROA comes from simulating the system backward in time from  $(90,0)$ . Points below this trajectory will eventually lead to failure by hitting the wall. The graphs on the right illustrate the approximated dynamics, the approximated constraints and the estimated ROA from the quartic (green line) and the constraints. In both cases, the estimated ROA covers 78% of real ROA, while the fixed center ellipsoid limited by the barrier captures only a fraction of it. Points **A** illustrate barrier crossing up to where  $\dot{B}_i$  becomes smaller than zero while **B** shows that part of the barrier where  $\dot{B}_i < 0$  can be crossed if they are behind other barriers.

Regions of attraction estimation using sums-of-squares scales polynomially in the number of state dimensions, with success stories up to about 14 dimensions [23].

#### ACKNOWLEDGMENTS

Elena Glassman was supported by the NDSEG and NSF graduate fellowships. Alexis Lussier Desbiens was supported by the NSERC and OAS, with additional support from DARPA DSO.

#### REFERENCES

- [1] J. H. McMasters and M. L. Henderson, "Low Speed Single Element Airfoil Synthesis," *Technical Soaring*, 1980.
- [2] J. M. Grasmeyer, M. T. Keennon, and Aerovironment, "Development of the Black Widow Micro Air Vehicle,"

- 39th AIAA Aerospace Sciences Meeting and Exhibit, October 2001.
- [3] Z. P. Bazant, "Size effect on structural strength: a review," *Archive of Applied Mechanics*, vol. 69, pp. 703–725, 1999.
- [4] R. Cory and R. Tedrake, "Experiments in Fixed-Wing UAV Perching," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, August 2008.
- [5] R. Cory, "Supermaneuverable perching," Ph.D. dissertation, MIT, June 2010.
- [6] A. Lussier Desbiens, A. Asbeck, and M. Cutkosky, "Landing, perching and taking off from vertical surfaces," *The International Journal of Robotics Research*, vol. 30, no. 3, pp. 355–370, 2011.
- [7] R. Tedrake, "LQR-Trees: Feedback motion planning on sparse randomized trees," in *Proceedings of Robotics:*

- Science and Systems (RSS)*, 2009, p. 8.
- [8] A. Goswami, B. Thuilot, and B. Espiau, "A study of the passive gait of a compass-like biped robot: symmetry and chaos," *International Journal of Robotics Research*, vol. 17, no. 12, 1998.
- [9] C. J. Tomlin, I. M. Mitchell, A. M. Bayen, and M. K. M. Oishi, "Computational techniques for the verification and control of hybrid systems," in *Multidisciplinary Methods for Analysis Optimization and Control of Complex Systems*, ser. Mathematics in Industry. Springer Berlin Heidelberg, 2005, pp. 151–175.
- [10] C. S. Hsu, "A theory of cell-to-cell mapping dynamical systems," *Journal of Applied Mechanics*, vol. 47, no. 4, pp. 931–939, 1980.
- [11] P. Tabuada, "Approximate symbolic models for verification," in *Verification and Control of Hybrid Systems*. Springer US, 2009, pp. 151–166.
- [12] P. A. Parrilo, "Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization," Ph.D. dissertation, California Institute of Technology, May 18 2000.
- [13] U. Topcu, A. Packard, and P. Seiler, "Local stability analysis using simulations and sum-of-squares programming," *Automatica*, vol. 44, no. 10, pp. 2669 – 2675, 2008.
- [14] S. Prajna, A. Jadbabaie, and G. Pappas, "A framework for worst-case and stochastic safety verification using barrier certificates," *IEEE Transactions on Automatic Control*, vol. 52, no. 8, pp. 1415 –1428, Aug 2007.
- [15] Motion Genesis, "Motion Genesis™," 2010. [Online]. Available: <http://www.motiongenesis.com/>
- [16] T. Kane and D. Levinson, *Dynamics, theory and applications*. McGraw Hill, 1985.
- [17] A. Asbeck, S. Kim, M. R. Cutkosky, W. R. Provancher, and M. Lanzetta, "Scaling Hard Vertical Surfaces with Compliant Microspine Arrays," *International Journal of Robotics Research*, vol. 25, no. 12, p. 14, September 2006.
- [18] H. K. Khalil, *Nonlinear Systems (3rd Edition)*, 3rd ed. Prentice Hall, Dec. 2001.
- [19] P. A. Parrilo, "Semidefinite programming relaxations for semialgebraic problems," *Mathematical Programming*, vol. 96, no. 2, pp. 293–320, 2003.
- [20] S. Canu, Y. Grandvalet, V. Guigue, and A. Rakotomamonjy, "SVM and kernel methods Matlab toolbox," Perception Systèmes et Information, INSA de Rouen, Rouen, France, 2005.
- [21] J. Löfberg, "YALMIP : A toolbox for modeling and optimization in MATLAB," in *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.
- [22] J. F. Sturm, "Using SeDuMi 1.02, a Matlab toolbox for optimization over symmetric cones," *Optimization Methods and Software*, vol. 11, no. 1-4, pp. 625 – 653, 1999.
- [23] U. Topcu, A. Packard, P. Seiler, and G. Balas, "Robust region-of-attraction estimation," *IEEE Transactions on Automatic Control*, vol. 55, no. 1, pp. 137 –142, Jan 2010.