AIAA SciTech Forum
January 3-7, 2022, San Diego, CA & Virtual
AIAA SCITECH 2022 Forum

10.2514/6.2022-2213

Robust Controller Synthesis for Vision-based Spacecraft Guidance and Control

Niyousha Rahimi∗, Shahriar Talebi∗, Aditya Deole∗, Mehran Mesbahi†
Dept. of Aeronautics & Astronautics, University of Washington, Seattle, Washington 98195, USA

Saptarshi Bandyopadhyay‡, and Amir Rahmani
Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California 91109, USA

This work develops a Robust Controller synthesis method for Vision-based Spacecraft (RCVS) guidance and control, integral to the robust autonomy framework for multi-spacecraft formation control and reconfiguration applications. The method is built around the use of a photo-realistic simulator, where a camera is deployed on a tracking spacecraft (ego) in order to observe an uncontrolled spacecraft (target) in a Low Earth Orbit (LEO). In this direction, the proposed approach performs the relative state (attitude and position) estimation of the target spacecraft using a Convolutional Neural Network (CNN). The state estimation error is then modeled and the corresponding error bounds are obtained around a nominal trajectory of the ego and target spacecraft. Next, this work proposes a linear matrix inequality (LMIs) based approach to controller synthesis, guaranteed to be robust against both model uncertainties and measurement errors, resulting from vision-based estimation. This controller is comprised of two distinct components, one synthesized based on the nominal trajectory, while the "robust" component corrects for deviations from the nominal trajectory. Finally, a tracking scenario that directly utilize the image data for spacecraft guidance and control, is presented to showcase the performance of the proposed robust autonomy framework.

I. Introduction

Close proximity operations for spacecraft have been extensively studied and tested in recent years [1–4]. Applications such as docking and on-orbit service heavily rely on information exchange between the cooperative spacecraft. Through an array of sensor measurements, these design processes rely on observer models like Extended Kalman Filter (EKF) to generate state estimations; as such, these techniques involve full or partial knowledge of target spacecraft [5]. The target may not cooperate in situations such as asteroid capture or orbital debris removal. In these cases, feature recognition over the target surface has been widely adopted. The problem is then broken into long-range and short-range estimation depending on the accuracy of feature recognition algorithms [6].

Incorporating insight from rich, perceptual sensing modalities such as cameras have the potential to convey more information than simple, single-output sensor devices. Despite the maturity of deep learning algorithms and CNN in many computer vision tasks, it has, only recently, become common for state estimation problems in space applications [7–11]. These new methods utilize high-resolution images to train a neural network for target-specific state estimation. The use of such tools can be challenging in the harsh lighting conditions and against a highly textured background (i.e., Earth) [12, 13]. Given enough data, deep learning algorithms can learn efficient models that map high-dimensional data to an array of outputs. However, modeling the uncertainties in the output is a challenging task. This becomes particularly crucial when the trained neural network is adopted as a perception module in the control loop for autonomous navigation—which requires specific safety guarantees.

In general, deep learning does not consider uncertainty representation in regression settings, and deep learning classification models often give normalized score vectors, which do not necessarily capture model uncertainty. To this end, the control community seeks to combine the best of learning with robust/optimal control for autonomous

∗Ph.D. Student, Robotics, Aerospace and Information Networks (RAIN) Lab, University of Washington, Seattle, WA 98195, USA
†Professor, Robotics, Aerospace and Information Networks (RAIN) Lab, University of Washington, Seattle, WA 98195, USA **AIAA Associate Fellow**
‡Robotics Technologist, Mobility and Robotic Systems Section, Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Drive, Pasadena, California 91109, USA.
Group Supervisor, Maritime and Multi-Agent Autonomy Group, Mobility and Robotic Systems Section, Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Drive, Pasadena, California 91109, USA. **AIAA Life Member**

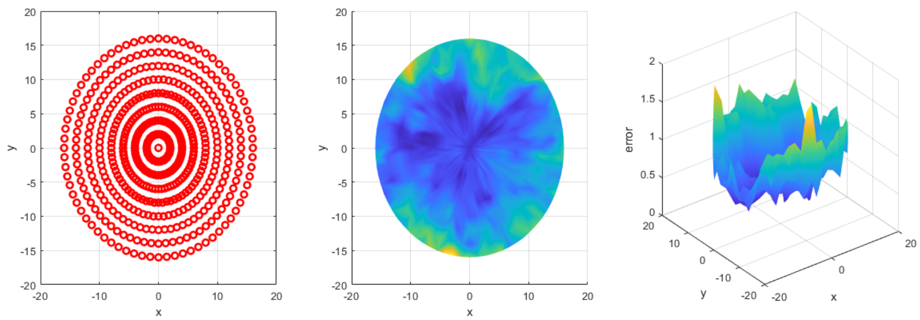


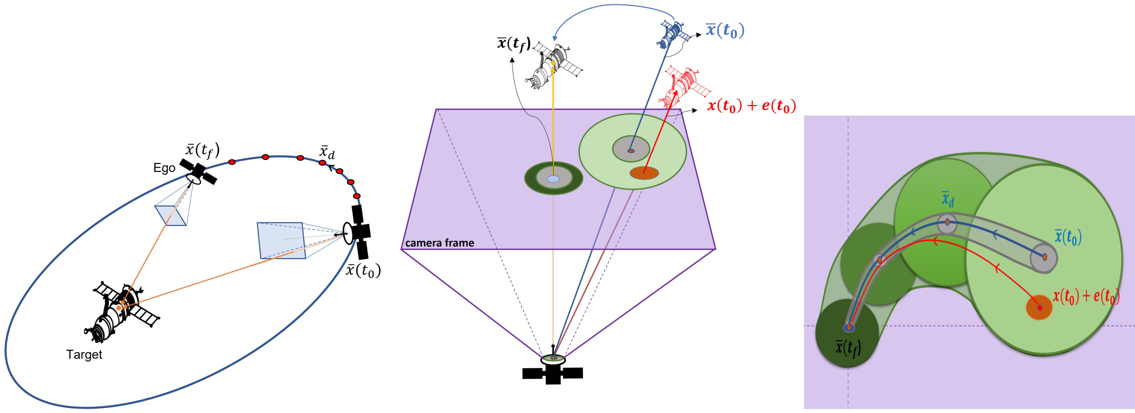
Fig. 1 The measurement error profile (center and right) showing the maximum errors around each training data point (shown in red in the left image), for a scenario comprised of a camera and a target—where all other influencing factors are held fixed except the camera orientation relative to the target (captured by $x - y$ coordinates in the camera frame). See [14] for more details.

navigation. In particular, in [15] the authors consider controlling a known linear dynamical system for which partial state information can only be extracted from complex observations. The approach is to design a virtual sensor by learning both a perception map (i.e., a map from (imagery) observations to a linear function of the state) and a bound on its estimation errors. Under suitable smoothness assumptions, "uniformly" bounded errors are guaranteed within a neighborhood of the training data. This uncertainty model allows synthesizing a robust controller that ensures the system does not deviate too far from the states visited during the training process. However, particularly in space applications, the information that can be extracted from imagery data may change drastically depending on the relative states of the system— e.g., see Figure 1 for an actual example of this phenomenon—. Therefore, a uniform analysis results in excessively conservative error modelings. Additionally, the error profile does not enjoy any particular distribution pattern and thus, assuming distribution profiles (such as Gaussian) would not be realistic. Here, instead, we aim to provide an error modeling that can capture its dependence on the relative states of the system and extracts guaranteed upper-bounds that are suitable for employment in the imminent robust control synthesis.

On a different note, explicit trajectory generation methods suffer from two fundamental drawbacks. First, they are inherently specific to the given problem data, and any changes necessitate a full resolution of the optimal control problem. Second, there is a lack of convergence guarantees for the solution of the corresponding nonconvex optimal control problems. No known algorithm can be formally guaranteed to solve a general nonconvex optimal control problem from an arbitrary initial guess. There is, therefore, no theoretical reason that could enable one to solve a nonconvex trajectory optimization problem in real-time —although certainly there is ample empirical data to support expectations of reliable local convergence [16]. It is natural, in response, to study alternative methods for which, the ability to provide a feasible trajectory in real-time can be theoretically established.

In this work, we aim to leverage contemporary techniques from both the machine learning and robust control literature to design a robust (nonlinear) controller– synthesizing a feedback signal directly from image sensors. For instance, consider the following scenario: A spacecraft (ego) has to navigate to a specific position and orientation with respect to another spacecraft (target) while keeping the target inside its camera field of view (see Figure 2a). This task requires estimation of the relative state of the target spacecraft in the ego's body frame, using a sequence of imagery observations. Our approach is to first, expand the analysis in [15] to design a virtual sensor by learning both a perception map (i.e., a map from observations to a linear function of the target state) and a "state-dependent" bound on its estimation errors. In contrast to [15], in this setup, the distance to the target and the sun's position dramatically affect the information that can be extracted from an image. Hence, we model the error of the composed perception process as a function of the target and ego's relative states.

Second, we utilize the implicit trajectory optimization methods to define a set of feasible trajectories in the state-and-control spaces. Then, based on a nominal trajectory and the Uniformly Ultimately Bounded (UUB) concept [17], we extend the invariant funnel synthesis technique [16] to a setup accounting for the state-dependent output measurement noise—where we can incorporate the error model and its upper bounds correspondingly. A schematic illustration of this idea is also depicted in Figures 2b and 2c. Finally, we arrive at the composite robust control synthesis that generates control signals directly from imagery data that can handle uncertainties in the (nonlinear) model as well as the (noisy) observations.



(a) (b) (c)

Fig. 2 The orbital trajectory of the ego around the target in its Hill's frame depicted in (a). In the ego's camera frame (purple plane), (b) illustrates the nominal trajectory starting at $\bar{x}(t_0)$ (in blue) with the target off of the center of the ego's camera, and terminal state $\bar{x}(t_f)$ (in black) shows the target aligned with the ego's camera center. The actual initial state $x(t_0)$ (shown in red) is contained in the invariant funnel at $t = 0$ (the light green ellipse). Finally, (c) shows the same nominal (blue) and actual (red) trajectories contained in the invariant funnel (green ellipses).

The rest of the paper is organized as follows. Section II outlines a representation of the nonlinear equations of motion by using the so-called structured nonlinearity. Also, the analysis for modeling the sensor uncertainties is laid out. Then, by introducing the concept of a quadratic funnel, a robust controller is designed—culminating in the framing of the quadratic funnel synthesis problem. The specific solution strategy is then presented in Section III. Next, in Section IV, the developed machinery is applied to a spacecraft target tracking problem. A nominal trajectory is designed, linearization of the system dynamics along this trajectory is outlined, and finally the performance of the general robust controller synthesis is showcased. The concluding remarks and future works are discussed in Section V.

II. Robust Controller Synthesis for Vision based Spacecraft Guidance and Control

A. Structured Nonlinear System Model with State Dependent Perception Error

In this work, we consider a problem setting in which the main uncertainty comes from the image sensor, generating observations that capture information about the state of a target relative to an ego spacecraft. Then, a specific (trained) "perception map" is utilized to recover the actual relative state of the system from the imagery data on the fly. Particularly, we define the (relative) state $x(t) \in \mathbb{R}^{n_x}$ (between ego and target spacecrafts) as

$$x(t) := x_{\text{ego}}(t) - x_{\text{target}}(t); \tag{1}$$

which follows a nonlinear dynamics

$$\dot{x}(t) = f(x(t), u(t)), \quad t \in \left[t_0, t_f\right], \tag{2}$$

$$z(t) = Q(x(t)), \tag{3}$$

where $u(t) \in \mathbb{R}^{n_u}$ denotes the control inputs, $z(t) \in \mathbb{R}^m$ represents sensory data as a function of the relative state, and $[t_0, t_f]$ is a prescribed time interval. The observed data $z(t) = Q(x_{\text{ego}}(t) - x_{\text{target}}(t))$ is then determined by the unknown generative model $Q : \mathbb{R}^{n_x} \longrightarrow \mathbb{R}^m$, which is nonlinear and potentially quite high dimensional. The task of the observation module is to detect the target and determine its relative position with respect to the ego spacecraft. Here, the observations $z(t)$ are the captured images and the map $Q : \mathbb{R}^{n_x} \longrightarrow \mathbb{R}^m$ generates these images as a function of the relative states, as in (3). For now, suppose there exists a *perception map* $p : \mathbb{R}^m \longrightarrow \mathbb{R}^l$ that imperfectly predicts the relative state $x(t)$, that is

$$p(z(t)) = Cx(t) + e_p(x(t)), \tag{4}$$

where $e_p(x(t))$ denotes the perception error which is *dependent* on the relative state. The details of this perception map and its error characterization is further discussed in the succeeding section.

The main control synthesis problem is composed of two main steps. First, a (nonlinear) controller, denoted by $\bar{u}(t)$, is designed which results in a nominal controlled state trajectory. Then, a correction input signal, denoted by $\xi(t)$, is synthesized that deals with both dynamics and measurement uncertainties rooted in the modeling and composed perception process, respectively. More precisely, we let $\{\bar{x}(t), \bar{u}(t)\}_{t=t_0}^{t_f}$ be the nominal trajectory that satisfies the dynamics (2) for some initial condition x_0, and define

$$\eta(t) := x(t) - \bar{x}(t) \quad \text{and} \quad \xi(t) := u(t) - \bar{u}(t). \tag{5}$$

By assuming that f is at least once differentiable, we can rewrite (2) in terms of the variables in (5) by using a first-order Taylor series expansion around the nominal trajectory as follows:

$$\dot{\eta}(t) = A(t)\eta(t) + B(t)\xi(t) + g(x(t), u(t)), \tag{6}$$

where $A(t)$ and $B(t)$ are the partial derivatives of f evaluated along the nominal trajectory, and the $g(x(t), u(t)$ represents the higher order (nonlinear) terms∗. The dynamical system in (6) can then be equivalently expressed using structured nonlinearities as follows

$$\begin{aligned}
\dot{\eta}(t) &= A(t)\eta(t) + B(t)\xi(t) + \sum_{i=1}^{N_w} E_i w_i(t) \\
h_i(t) &= F_i\eta(t) + G_i\xi(t), && i = 1, \ldots, N_w \\
w_i(t) &= \phi_i\left(h_i(t)\right), && i = 1, \ldots, N_w
\end{aligned}$$

where the pairs $(h_i, w_i) \in \mathbb{R}^{n_{h,i}} \times \mathbb{R}^{n_{w,i}}$ capture the higher order terms through the nonlinear functions ϕ_i. The details of this transformation are laid out in [20], and the constant matrices $F_i \in \mathbb{R}^{n_h \times n_x}$, $G_i \in \mathbb{R}^{n_h \times n_u}$ and $E_i \in \mathbb{R}^{n_x \times n_w}$ serve as nonlinear inputs/outputs channel selectors.

Finally, based on the nominal trajectory $\{\bar{x}(t), \bar{u}(t)\}_{t=t_0}^{t_f}$, we can define the output of the system using the composed perception process in (4) as

$$y(t) := p(z(t)) - C\bar{x}(t) = C\eta(t) + e(\eta(t)), \tag{7}$$

where $e(\eta(t))$ denotes the state-dependent measurement noise—to be characterized later in Section II.B. Hence, we arrive at the following (nonlinear) system dynamics and (state-dependent) noisy observations:

$$\begin{aligned}
\dot{\eta}(t) &= A(t)\eta(t) + B(t)\xi(t) + \sum_{i=1}^{N_w} E_i w_i(t) \\
h_i(t) &= F_i\eta(t) + G_i\xi(t), && i = 1, \ldots, N_w \\
w_i(t) &= \phi_i\left(h_i(t)\right), && i = 1, \ldots, N_w, \\
y(t) &= C\eta(t) + e(\eta(t)).
\end{aligned} \tag{8}$$

Depending on the the perception module, it is reasonable to assume that $C = I$ in most of applications which only simplifies the presentation of our proposed methodology, however, we provide clarifications whenever it is necessary.

B. Modelling Sensor Uncertainties

In order to obtain a perception map p that can recover the relative state "x" from the imagery data "z", we suppose access to initial training dataset $\mathcal{S}_0 = \{(x_d, z_d)\}_{d=1}^{N_0}$ which are collected *densely enough* from the nominal trajectory $\{\bar{x}(t), \bar{u}(t)\}_{t=t_0}^{t_f}$. This dataset is used to learn a perception map via any wide variety of traditional supervised methods [14]. To accurately model the uncertainties in the output measurements (7)—which will be subsequently used in the robust control synthesis—we estimate the safe regions around the nominal trajectory in terms of parameters learned from a second (testing) dataset. Next, we define the regions of the state space within which the sensing is reliable. To this end, we consider the following local assumptions:

∗An alternative dynamics modeling and online control purely based on data will be considered as a future direction, see e.g., [18, 19]

Assumption 1. *The mapping $p \circ Q - C$ is locally s-slope bounded with a radius of r around each training data points, i.e., for each $(x_d, z_d) \in \mathcal{S}_0$, there exists a bounded real number $s = s(x_d)$ such that*

$$\|p \circ Q(x) - Cx - (p \circ Q(x_d) - Cx_d)\| \leq s(x_d) \|(x - x_d)\|, \quad \forall x \in B_r(x_d). \tag{9}$$

Assumption 2. *The training error at each point of the training dataset is bounded, i.e., for each $(x_d, z_d) \in \mathcal{S}_0$, there exists a bounded real number $\gamma = \gamma(x_d)$ such that*

$$\|p(z_d) - Cx_d\| \leq \gamma(x_d).$$

Note that the modeling assumptions above are completely local which enables us to utilize the perception module to its best extent and avoid using the *uniform* worst-case-scenario modeling. Next, the regions of the state space within which the sensing is reliable can then be described using a safe set which approximates sub-level sets of the norm of the state-dependent error function $e(x) = p(q(x)) - Cx$ around the nominal trajectory. More precisely, given $r > 0$, we define

$$\mathcal{X}_r(\bar{x}(t)) := \left\{ x \in B_r(\bar{x}(t)) \,\middle|\, \|p(\widetilde{z}_d) - C\widetilde{x}_d\| + s(t) \|\overline{x}(t) - \widetilde{x}_d(t)\| \leq l(t), \quad \widetilde{x}_d(t) := \arg\min_{x_d \in \mathcal{S}_0} \|x - x_d\| \right\}, \tag{10}$$

where $\widetilde{z}_d(t) := Q(\widetilde{x}_d(t))$, $\gamma(t) := \gamma(\widetilde{x}_d(t))$, and

$$s(t) := s(\widetilde{x}_d(t)), \quad l(t) := rs(t) + \gamma(t). \tag{11}$$

Note that we overload the notation to define $s(t)$, which is the maximum slope bound of the mapping $p \circ Q - C$ at all $x \in B_r(\widetilde{x}_d)$. Since the nominal trajectory $\{\bar{x}(t), \bar{u}(t)\}_{t=t_0}^{t_f}$ is fixed and the training data $\mathcal{S}_0$ is collected from this trajectory, we are able to define the safe set $\mathcal{X}_r(t) = \mathcal{X}_r(\bar{x}(t))$ as a function of time around the nominal trajectory. Further, note that parameters $s(x_d)$ can be learned using a second dataset (see [14] for more details). Finally, we can precisely define how densely the nominal trajectory should be sampled for the training dataset as follows:

Definition 1. We say that the training dataset $\mathcal{S}_0 = \{(x_d, z_d)\}_{d=1}^{N_0}$ is *sampled densely enough* from nominal trajectory $\{\bar{x}(t), \bar{u}(t)\}_{t=t_0}^{t_f}$ if there exists a positive radius $r > 0$ such that for any $x \in B_r(\bar{x}(t))$ and any $t \in [t_0, t_f]$, we have $\|\bar{x}(t) - \widetilde{x}_d(t)\| \leq r$ where $\arg\min_{x_d \in \mathcal{S}_0} \|x - x_d\|$.

The error characteristics considered in our approach is different from that of [15] as each training data point x_d is associated with different parameters $\gamma(x_d)$ and $s(x_d)$, and we parametrized the safe set around the nominal trajectory as a function of time. The rationale behind this modeling is that here, the error of the composed perception process depends on the relative state between ego and target spacecrafts, certainly because, as we get farther away from the target, the amount of information that can be extracted from the images decreases significantly. Moreover, environmental factors such as target reflection or the sun's position can also dramatically affect the perception error. Hence, the training error increases accordingly and in order to model this phenomenon, we assume that the training error is a function of the training points as described in Assumptions 1 and 2.

Now, we are well-equipped to characterize the perception error—altering the output of the system as modeled in (7)—in the following lemma.

Lemma 3. *Suppose the training dataset is sampled densely enough from the nominal trajectory (per Definition 1) with a radius r and Assumptions 1 and 2 hold with the same radius. Then, for any $x(t) \in \mathcal{X}_r(\bar{x}(t))$, the perception error is locally bounded with*

$$\|e(\eta(t))\| \leq s(t)\|\eta(t)\| + l(t).$$

Proof. For any $x(t) \in \mathcal{X}_r(\bar{x}(t))$ (corresponding to a sensory data $z(t) = Q(x(t))$) such that $x(t) \in B_r(\bar{x}(t))$, consider $(\widetilde{x}_d(t), \widetilde{z}_d(t)) \in \mathcal{S}_0$ and observe that

$$\begin{aligned}\|p(z) - Cx\| &= \|p(z) - Cx - (p(\widetilde{z}_d) - C\widetilde{x}_d) + p(\widetilde{z}_d) - C\widetilde{x}_d\| \\ &\leq s(x_d) \|(x - \widetilde{x}_d)\| + \|p(\widetilde{z}_d) - C\widetilde{x}_d\|,\end{aligned}$$

where the last inequality follows from the local slope bound assumption. Recall that $\eta(t) := x(t) - \bar{x}(t)$ and the nominal trajectory $\{\bar{x}(t), \bar{u}(t)\}_{t=t_0}^{t_f}$ is known. Since the training dataset $\mathcal{S}_0 = \{(x_d, z_d)\}_{d=1}^{N_0}$ is sampled densely enough from the nominal trajectory, for any $(x(t), z(t))$ with $x(t) \in \mathcal{X}_r(\bar{x}(t))$ we can upper bound the perception error in (8) as follows

$$\begin{aligned}\|e(\eta(t))\| &\leq s(t)\| \left(x(t) - \bar{x}(t) + \bar{x}(t) - \widetilde{x}_d(t)\right) \| + \|p\left(Q\left(\widetilde{x}_d\right)\right) - C\widetilde{x}_d(t)\| \\ &\leq s(t)\|\eta(t)\| + s(t)\| \left(\bar{x}(t) - \widetilde{x}_d(t)\right) \| + \|p\left(Q\left(\widetilde{x}_d\right)\right) - C\widetilde{x}_d(t)\| \leq s(t)\|\eta(t)\| + l(t),\end{aligned}$$

where the last inequality follows by definition. □

Finally, note that as we have a more densely sampled nominal trajectory (i.e., as $r > 0$ is smaller,) the affine upper-model of the noise will have smaller shift $l(t)$, however, the slope-bound and training errors are inherent to the learning problem at hand and does not necessarily improve by just sampling more densely.

C. Robust Controller Synthesis

In this section we explicate how the problem reformulation and error modeling presented so far, can be utilized for synthesizing a robust controller that overcomes the uncertainties inherent to both the dynamics model and the perception process. First, as a result of Lemma 3, we can consider the uncertain nonlinear system in (8) with the following perception error

$$\|e(\eta(t))\| \leq s(t)\|\eta(t)\| + l(t). \tag{12}$$

We consider the class of controllers that, at each iteration t, can synthesis a correction signal $\xi(t) = \mathbf{K}(t)y(t)$ for some matrix-valued function of time $\mathbf{K}(t) \in \mathbb{R}^{n_u \times n_y}$. The closed-loop system representation obtains the following form:

$$\begin{aligned}\dot{\eta}(t) &= \left(A(t) + B(t)\mathbf{K}(t)\right)\eta(t) + B(t)\mathbf{K}(t)e\left(\eta(t)\right) + \sum_{i=1}^{N_w} E_i w_i(t) \\ h_i(t) &= \left(F_i + G_i\mathbf{K}(t)\right)\eta(t) + G_i\mathbf{K}(t)e\left(\eta(t)\right), \qquad & i = 1, \cdots, N_w \\ w_i(t) &= \phi_i\left(h_i(t)\right), \qquad & i = 1, \cdots, N_w, \\ \|e(\eta(t))\| &\leq s(t)\|\eta(t)\| + l(t),\end{aligned} \tag{13}$$

As discussed in the previous section, the perception error is bounded as (12) in a vicinity of the nominal trajectory. We use this fact to model the term $\mathbf{K}(t)e(\eta(t))$ in the dynamics of $\eta(t)$ as a structured nonlinearity. In fact, we aim to design a feedback synthesis procedure that can handle any possible perception error $e(t)$ that satisfies the upper bound model $\|e(t)\| \leq s\|\eta(t)\| + l(t)$, including the realized error which inherently may depend on $\eta(t)$. The details are provided in the following subsection.

1. Modeling the Perception Error in Feedback as Added Nonlinearity

Herein, we propose a specific modeling of the the perception error that eventually enables us to derive at an LMI formulation for synthesizing the feedback gain K. It is worth noting that the direct approach for considering the error in measurements results in nonlinear inequalities with no known LMI equivalent. In particular, we consider the error $e(t)$ as a nonlinear channel and model the effects of any such error (including the perception error $e(\eta(t))$)—that satisfies the upper-bound model $\|e(t)\| \leq s\|\eta(t)\| + l(t)$—as

$$w_e(t) \coloneqq \widetilde{\phi}\left(e(t)\right) = \mathbf{K}(t)\, e(\eta(t)), \quad \text{for some} \quad \widetilde{\phi} : \mathbb{R}^{n_y} \to \mathbb{R}^{n_u}. \tag{14}$$

By considering $e(t)$ as a nonlinear input channel, we then need to update the nonlinear input channel selectors in $h(t)$

$$h_i(t) = \left(F_i + G_i\mathbf{K}(t)\right)\eta(t) + \widetilde{G}_i e(t), \qquad i = 1, \cdots, N_w \tag{15}$$

where the constant matrix $\widetilde{G}_i \in \mathbb{R}^{n_x \times n_y}$ serves as nonlinear input channel selector for $e(t)$. We then stack each of the ϕ_i to construct the function $\phi : \mathbb{R}^{n_h} \to \mathbb{R}^{n_w}$, and rewrite (13) together with (14) and (15) as

$$\begin{aligned}\dot{\eta}(t) &= A_{cl}(t)\eta(t) + B(t)w_e(t) + Ew(t), \\ h(t) &= F_{cl}(t)\eta(t) + \widetilde{G}e(t), \\ w(t) &= \phi(h(t)), \\ w_e(t) &= \widetilde{\phi}\left(e(t)\right)\end{aligned} \tag{16}$$

where

$$w = \begin{bmatrix} w_1 \\ \vdots \\ w_{N_w} \end{bmatrix}, \quad h = \begin{bmatrix} h_1 \\ \vdots \\ h_{N_w} \end{bmatrix}, \quad F = \begin{bmatrix} F_1 \\ \vdots \\ F_{N_w} \end{bmatrix}, \quad G = \begin{bmatrix} G_1 \\ \vdots \\ G_{N_w} \end{bmatrix}, \quad \widetilde{G} = \begin{bmatrix} \widetilde{G}_1 \\ \vdots \\ \widetilde{G}_{N_w} \end{bmatrix}, \quad E = \begin{bmatrix} E_1 & \cdots & E_{N_w} \end{bmatrix},$$

such that $w \in \mathbb{R}^{n_w}$ and $h \in \mathbb{R}^{n_h}$, $F \in \mathbb{R}^{n_h \times n_x}, G \in \mathbb{R}^{n_h \times n_u}, \widetilde{G} \in \mathbb{R}^{n_h \times n_y}$, and $E \in \mathbb{R}^{n_x \times n_w}$ with $n_w = \sum_{i=1}^{N_w} n_{w,i}$ and $n_h = \sum_{i=1}^{N_w} n_{h,i}$, and $A_{\mathrm{cl}}(t) := A(t) + B(t)\mathbf{K}(t)$ and $F_{\mathrm{cl}}(t) := F + G\mathbf{K}(t)$.

2. Funnel Synthesis

The funnel synthesis techniques are based on the notion of quadratic stability as defined in [21] and [22–24], the latter of which offers necessary and sufficient conditions for stability based on quadratic Lyapunov functions. As described in [16], funnel synthesis makes use of a Lyapunov function (as opposed to the maximum principle-based techniques) to seek out nearby feasible trajectories. Following is a formal definition of a funnel.

Definition 2. (Funnel). A funnel, denoted by $\mathcal{F}(t)$, is a time-varying set in state and control space that is both invariant and contained inside a feasible region.

The invariance property of a funnel means that if a particular initial condition is inside the entry of the funnel (at some initial time t_0), then the entire subsequent trajectory remains inside the funnel as well. Stated mathematically, if $(\eta(t_0), \xi(t_0)) \in \mathcal{F}(t_0)$ then $(\eta(t), \xi(t)) \in \mathcal{F}(t)$ for all $t \geq t_0$. The goal is to seek the largest possible funnel that satisfies input and state constraints. This allows us to implicitly define a large family of trajectories by using the functions that define the funnel thereby providing the ability to guarantee the availability of a feasible trajectory over a larger region of parameter variations. Funnel synthesis guarantees convergence behavior and is real-time capable on computationally constrained hardware.

Particularly, consider the scalar-valued function $V : \mathbb{R}^{n_x} \to \mathbb{R}$ defined by

$$V(\eta(t)) = \eta(t)^\top Q(t)^{-1} \eta(t) \tag{17}$$

where $Q(t) \in \mathbb{S}^{n_x}_{++}$ is a matrix-valued function of time whose range space lies in the set of positive definite matrices. As a result, we have $V(\eta(t)) > 0$ for all $t \in [t_0, t_f]$ whenever $\eta(t) \neq 0$.

Having introduced each of the time-varying terms, we henceforth omit the argument of time "t", whenever possible. The 1-level set of $V(\eta(t))$ is the set of states that satisfy the quadratic inequality $\eta^\top Q^{-1} \eta \leq 1$, which is also the equation of a non-degenerate n_x-dimensional ellipsoid. We denote the ellipsoid defined by the positive definite matrix Q and centered at the origin as

$$\mathcal{E}_Q := \left\{ \eta \in \mathbb{R}^{n_x} \mid \eta^\top Q^{-1} \eta \leq 1 \right\} = \left\{ Q^{1/2} v \mid \|v\|_2 \leq 1 \right\} \tag{18}$$

The assumption that $\xi = Ky$ used to form the closed-loop system (16) thus results in the following implication:

$$\text{if} \quad y \in \mathcal{E}_{Q_y} \quad \Rightarrow \quad \xi \in \mathcal{E}_{K Q_y K^\top} \tag{19}$$

A fact that can be proven easily via Schur complements when K is full row-rank [20]. Before we proceed, we need to drive an upper bound for Q_y, given the information about the perception error.

Theorem 4. *If $\eta \in \mathcal{E}_Q$ and the perception error is bounded $\|e\| \leq s\|\eta\| + l$, then $y \in \mathcal{E}_{Q_y}$, where*

$$Q_y = (1 + s(1+\rho))^2 \, Q, \qquad Q_s := s^2 Q, \quad \rho := \sqrt{l \lambda_{\max}\left(Q_s^{-1}\right)} \tag{20}$$

and $\lambda_{\max}\left(Q_s^{-1}\right)$ is the largest eigenvalue of Q_s^{-1}.

Proof. Let $e = e_\eta + r$, where $\|e_\eta\| \leq s\|\eta\|$ and $\|r\| \leq l$. Define $Q_s := s^2 Q$, then by the Schur complement it follows that $e_\eta \in \mathcal{E}_{Q_s}$. We further claim that

$$e \in \mathcal{E}_{Q_e}, \quad \text{where} \quad Q_e := (1+\rho)^2 \, Q_s. \tag{21}$$

The proof utilizes the fact that for any positive definite matrix P and any vectors x, y, we have $2x^\top P y \leq \frac{1}{a} x^\top P x + a y^\top P y$ for all $a > 0$. In particular, for any $\hat{\rho}$, we have

$$e^\top {Q_e}^{-1} e = (e_\eta + r)^\top \left((1+\rho)^2 Q_s \right)^{-1} (e_\eta + r) \leq (1+\rho)^{-2} (1+\hat{\rho}) e_\eta^\top Q_s^{-1} e_\eta + (1+\rho)^{-2} \left(1 + \frac{1}{\hat{\rho}} \right) r^\top Q_s^{-1} r$$

Let $\hat{\rho} = \rho$ and note that since $\|r\| \leq l$, then $r^\top Q_s^{-1} r \leq l\lambda_{\max}\left(Q_s^{-1}\right)$. Hence we have $e^\top Q_e e \leq 1$ and the proof of the claim is complete. Similarly, for $y = \eta + e$ we have

$$y^\top {Q_y}^{-1} y = (\eta + e)^\top Q_y (\eta + e) \leq (1 + s(1+\rho))^{-2}(1+\hat{\rho})\,\eta^\top Q^{-1}\eta + (1+s(1+\rho))^{-2}\left(1 + \frac{1}{\hat{\rho}}\right) e^\top Q^{-1} e$$

Let $\hat{\rho} = s(1+\rho)$. The proof of the theorem follows from the over-approximation of Q_e, where $e^\top {Q_e}^{-1} e = e^\top \left(s(1+\rho)^2 Q\right)^{-1} e \leq 1$. □

Suppose that $\mathcal{X}_\mathcal{F} \in \mathbb{R}^{n_x}$ and $\mathcal{U}_\mathcal{F} \in \mathbb{R}^{n_u}$ are the (possibly nonconvex) sets of feasible state and control vectors. Using these feasible sets, we formally define a quadratic invariant funnel as follows.

Definition 3. (Quadratic Funnel). A quadratic funnel, $\mathcal{F}$, is a subset of the Cartesian product of the feasible state space $\mathcal{X}_\mathcal{F}$ and control space $\mathcal{U}_\mathcal{F}$ (so-called feasible state-and-control space) that is parameterized by time-varying positive definite matrices $Q \in \mathbb{S}_{++}^{n_x}$, $Q_y \in \mathbb{S}_{++}^{n_y}$ and a time-varying matrix $K \in \mathbb{R}^{n_u \times n_y}$. Specifically, we define

$$\mathcal{F} := \mathcal{E}_Q \times \mathcal{E}_{KQ_yK^\top} \subseteq \mathcal{X}_\mathcal{F} \times \mathcal{U}_\mathcal{F}. \tag{22}$$

3. Invariance of Quadratic Funnel in Presence of Measurement Noise

In this work, we have incorporated the funnel synthesis techniques to provide one (or more) functions that implicitly define a set of trajectories in the state-and-control space. As the closed-loop system— described in (16)—is obtained from an output feedback controller with (state-dependent) measurement and process noise, one has to consider an appropriate stability criterion. Hence, we require that any proposed controller renders a UUB† closed-loop system. We utilized Theorem 4.18 in [17] to design a controller **K** such that the closed loop system (16) is UUB, in the presence of state-dependent noise and system nonlinearity, and starting anywhere in an invariant funnel. A sufficient condition to achieve this goal in view of the dynamics in (16) is detailed as follows:

$$\alpha_1(\|\eta(t)\|) \leq V(\eta(t)) \leq \alpha_2(\|\eta(t)\|), \qquad \frac{\partial V}{\partial t} + \frac{\partial V}{\partial \eta}\dot{\eta}(t) \leq -V(\eta(t)), \qquad \forall\, t \in [t_0, t_f], \qquad \forall \eta \ni \|C_\mu \eta\| \geq \mu \geq 0,$$
$$\forall h \in \Omega, \quad \text{and} \quad \forall e \in \mathcal{E}_{Q_e} \ni \|e(t)\| \leq s(t)\|\eta(t)\| + l(t), \tag{23}$$

where μ denotes the time-varying UUB parameter. Note that the quadratic funnel's invariance property means that $\eta(t) \in \mathcal{E}_Q$ for all times $t \in [t_0, t_f]$. This implies that the vector h, which is the input to the nonlinear terms in system (13), must satisfy the set inclusion

$$h \in \Omega, \quad \Omega = \mathcal{E}_{F_{cl}QF_{cl}^\top} \oplus \mathcal{E}_{\widetilde{G}Q_e\widetilde{G}^\top} \tag{24}$$

where $\oplus$ is the Minkowski sum between two sets. In order to utilized the condition (23) for synthesizing a quadratic funnel, we need to express the nonlinear expressions "$w = \phi(h), \quad \forall h \in \Omega$" and "$w_e = \widetilde{\phi}(e), \quad \forall\, e \in \mathcal{E}_{Q_e}$"—appearing in (16)—in such a way that it is consistent with the quadratic form of the energy function $V(\eta(t))$. To this end, we use the idea of local multiplier matrices as also adopted in [16].

Definition 4. Consider any nonlinear map $\phi : \mathbb{R}^{n_h} \to \mathbb{R}^{n_w}$ that sends $h \to \phi(h)$. We say a symmetric matrix $M \in \mathbb{S}^{(n_h+n_w)}$ is a *local multiplier matrix* for ϕ over the set Ω if

$$\mathcal{T}(h) := \begin{bmatrix} h \\ \phi(h) \end{bmatrix}^\top M \begin{bmatrix} h \\ \phi(h) \end{bmatrix} \geq 0, \quad \forall\, h \in \Omega. \tag{25}$$

Furthermore, we denote the set of local multiplier matrices for ϕ over the set Ω by

$$\mathcal{M}_{\phi,\Omega} := \left\{ M \in \mathbb{S}^{(n_h+n_w)} \mid \mathcal{T}(h) \geq 0, \quad \forall\, h \in \Omega \right\}. \tag{26}$$

†See Definition 4.6 in [17] for more details on UUB concept.

Similarly we define the set of local multiplier matrices for $\widetilde{\phi}(e)$, where M_e denotes a *local multiplier matrix* for $\widetilde{\phi}$ over the set Q_e:

$$\mathcal{T}_e(e) := \begin{bmatrix} e \\ \widetilde{\phi}(e) \end{bmatrix}^\top M_e \begin{bmatrix} e \\ \widetilde{\phi}(e) \end{bmatrix} \geq 0, \quad \forall\, e \in \mathcal{E}_{Q_e}. \tag{27}$$

Finally, the following theorem provides sufficient conditions that guarantees the invariance property of a quadratic funnel in the presence of both measurement and modeling uncertainties.

Theorem 5. *The output controller of the form $\xi(t) = \mathbf{K}(t)y(t)$ satisfies the feasibility conditions (23) if there exists $Q > 0,\ Y \in \mathbb{R}^{n_u \times n_x}$, and multiplier matrices $M \in \mathcal{M}_{\phi,\Omega}$, $M_e \in \mathcal{M}_{\widetilde{\phi},\mathcal{E}_{Q_e}}$ and scalars $\tau, \tau_e, \alpha, \lambda > 0$ such that for all $t \in [t_0, t_f]$,*

$$\begin{bmatrix} \mathbf{Q}A^\top + A\mathbf{Q} - \dot{\mathbf{Q}} + \mathbf{Q} + T_1 & E + \tau \mathbf{F}_{cl}^\top M_{12} & B & \tau \mathbf{F}_{cl}^\top M_{11}\widetilde{G} & 0 \\ E^\top + \tau M_{12}^\top \mathbf{F}_{cl} & \tau M_{22} & 0 & \tau M_{12}^\top \widetilde{G} & 0 \\ * & * & [& T_2 &] \end{bmatrix} \leq 0 \tag{28}$$

where

$$T_1 = \tau\ \mathbf{F}_{cl}^\top M_{11}\ \mathbf{F}_{cl} + \alpha \mathbf{Q} C_\mu^\top C_\mu \mathbf{Q} + \lambda s \mathbf{Q}\mathbf{Q}, \qquad \mathbf{F}_{cl} = F\mathbf{Q} + G\mathbf{Y}$$
$$T_2 = \begin{bmatrix} \tau_e M_{e22} & \tau_e M_{e12}^\top & 0 \\ \tau_e M_{e12} & \tau \widetilde{G}^\top M_{11} \widetilde{G} + \tau_e M_{e11} - \lambda I & 0 \\ 0 & 0 & -\alpha\mu + \lambda R \end{bmatrix} \tag{29}$$

Proof. Using local multiplier matrices and because $V(\eta(t))$ is not an explicit function of t, we can rewrite the condition in (23) as

$$\dot{V}(\eta(t)) + V(\eta(t)) \leq 0, \quad \forall t \in [t_0, t_f], \quad \begin{bmatrix} h \\ w \end{bmatrix}^\top M \begin{bmatrix} h \\ w \end{bmatrix} \geq 0,$$
$$\begin{bmatrix} e \\ w_e \end{bmatrix}^\top M_e \begin{bmatrix} e \\ w_e \end{bmatrix} \geq 0, \quad \|C_\mu \eta(t)\| \geq \mu \geq 0, \quad \|e(\eta(t))\| \leq s(t)\|\eta(t)\| + l(t), \tag{30}$$

for some $M \in \mathcal{M}_{\phi,\Omega}$ and $M_e \in \mathcal{M}_{\widetilde{\phi},\mathcal{E}_{Q_e}}$. Expanding this condition in terms of the closed-loop system (16) leads to

$$\begin{bmatrix} \eta \\ w \\ w_e \end{bmatrix}^T \begin{bmatrix} \mathcal{A}^T Q^{-1} + Q^{-1}\mathcal{A} - Q^{-1}\dot{Q}Q^{-1} + Q^{-1} & Q^{-1}E & Q^{-1}B \\ E^\top Q^{-1} & 0 & 0 \\ B^\top Q^{-1} & 0 & 0 \end{bmatrix} \begin{bmatrix} \eta \\ w \\ w_e \end{bmatrix} \leq 0$$

for all η, w and w_e that satisfy

$$\begin{bmatrix} \eta \\ e \\ w \end{bmatrix}^\top \begin{bmatrix} F_{cl}^\top M_{11} F_{cl} & F_{cl}^\top M_{11}\widetilde{G} & F_{cl}^\top M_{12} \\ \widetilde{G}^\top M_{11} F_{cl} & \widetilde{G}^\top M_{11}\widetilde{G} & \widetilde{G}^\top M_{12} \\ M_{12}^\top F_{cl} & M_{12}^\top \widetilde{G} & M_{22} \end{bmatrix} \begin{bmatrix} \eta \\ e \\ w \end{bmatrix} \geq 0, \quad \begin{bmatrix} e \\ w_e \end{bmatrix}^\top \begin{bmatrix} M_{e11} & M_{e12} \\ M_{e12}^\top & M_{e22} \end{bmatrix} \begin{bmatrix} e \\ w_e \end{bmatrix} \geq 0$$
$$\begin{bmatrix} \eta \\ 1 \end{bmatrix}^\top \begin{bmatrix} C_\mu^\top C_\mu & 0 \\ 0 & -\mu \end{bmatrix} \begin{bmatrix} \eta \\ 1 \end{bmatrix} \geq 0, \quad \begin{bmatrix} \eta \\ e \\ 1 \end{bmatrix}^\top \begin{bmatrix} sI & 0 & 0 \\ 0 & -I & 0 \\ 0 & 0 & R \end{bmatrix} \begin{bmatrix} \eta \\ e \\ 1 \end{bmatrix} \geq 0 \tag{31}$$

where $\mathcal{A} = A + B\mathbf{K}$ and $F_{cl} = F + G\mathbf{K}$. By using the S-procedure, the last condition holds if and only if there exists scalars $\tau \geq 0, \alpha \geq 0$ and $\lambda \geq 0$ such that

$$\begin{bmatrix} \mathcal{A}^\top \mathbf{Q}^{-1} + \mathbf{Q}^{-1}\mathcal{A} - \mathbf{Q}^{-1}\dot{\mathbf{Q}}\mathbf{Q}^{-1} + \mathbf{Q}^{-1} + T_1 & \mathbf{Q}^{-1}E + \tau F_{cl}^\top M_{12} & \mathbf{Q}^{-1}B & \tau F_{cl}^\top M_{11}\widetilde{G} & 0 \\ E^\top Q^{-1} + \tau M_{12}^\top F_{cl} & \tau M_{22} & 0 & \tau M_{12}^\top \widetilde{G} & 0 \\ * & * & [& T_4 &] \end{bmatrix} \leq 0 \tag{32}$$

where

$$T_1 = \tau\ F_{cl}^\top M_{11}\ F_{cl} + \alpha C_\mu^\top C_\mu + \lambda s I, \quad T_2 = \begin{bmatrix} \tau_e M_{e22} & \tau_e M_{e12}^\top & 0 \\ \tau_e M_{e12} & \tau \widetilde{G}^\top M_{11}\widetilde{G} + \tau_e M_{e11} - \lambda I & 0 \\ 0 & 0 & -\alpha\mu + \lambda R \end{bmatrix} \tag{33}$$

Pre- and post-multiplying (28) by $\mathbf{diag}\{Q, I, I, I\}$, will complete the proof.

□

4. Feasibility of Quadratic Funnel in Presence of Measurement Noise

We now discuss the conditions that are required for a quadratic funnel to be contained in a given feasible region. As it was discussed in [16], the goal is to find the "maximum quadratic funnel", denoted by $\mathcal{F}_{\max}$, that satisfies state and control constraints. Suppose that $\mathcal{X}_{\mathcal{F}} \subset \mathbb{R}^{n_x}$ and $\mathcal{U}_{\mathcal{F}} \subset \mathbb{R}^{n_u}$ are the (possibly nonconvex) feasible subsets of the state-and-control space. We assume that the nominal trajectory satisfies $\bar{x} \in \text{int } \mathcal{X}_{\mathcal{F}}$ and $\bar{u} \in \text{int } \mathcal{U}_{\mathcal{F}}$, or, in words, the nominal trajectory lies in the strict interior of the two corresponding feasible subsets. This assumption is necessary to avoid degenerate solutions during funnel synthesis procedure. Note that, this assumption does not preclude nominal trajectories that activate a constraint; it merely suggests that the constraint sets used for funnel synthesis are slightly more relaxed than those imposed during nominal trajectory synthesis, if necessary. This can be easily done by inflation/deflation of the feasibility subsets. Also, the set that determines the maximum size of the funnel at the final time $t = t_f$, is denoted by $\mathcal{X}_{\mathcal{F}}^f \subset \mathbb{R}^{n_x}$. Then, the maximum quadratic funnel satisfies

$$\mathcal{F}_{\max} = \mathcal{E}_{Q_{\max}} \times \mathcal{E}_{R_{\max}} \quad \text{and} \quad \mathcal{E}_{Q_{\max}} \subseteq \mathcal{X}_{\mathcal{F}}, \quad \mathcal{E}_{R_{\max}} \subseteq \mathcal{U}_{\mathcal{F}} \tag{34}$$

where $Q_{\max} \in \mathbb{S}_{++}^{n_x}$ and $R_{\max} \in \mathbb{S}_{++}^{n_u}$. The, inclusion in the terminal set $\mathcal{X}_{\mathcal{F}}^f$ can be guaranteed by ensuring that $\mathcal{E}_{Q_{\max}} \subseteq \mathcal{X}_{\mathcal{F}}^f$ when $t = t_f$.

To compute the matrix $Q_{\max}$ at any time $t < t_f$, we seek to find the best quasi-polytopic approximation of $\mathcal{X}_{\mathcal{F}}$ in the vicinity of the nominal trajectory. To this end, following the analysis presented in [16], we assume that the set $\mathcal{X}_{\mathcal{F}}$ can be expressed as

$$\mathcal{X}_{\mathcal{F}} = \{x \mid h_i(x) \le 0, i = 1, \dots, m_x, \|x\|_2 \le x_{\max}\} \tag{35}$$

and the quasi-polytopic approximation of $\mathcal{X}_{\mathcal{F}}$ is

$$\mathcal{P}_{\bar{x}} = \left\{x \mid a_i^\top x \le b_i,\ i = 1, \dots, m_x\right\} \cap \{x \mid \|x\|_2 \le x_{\max}\} \tag{36}$$

where the data $\{a_i, b_i\}_{i=1}^{m_x}$ are first-order approximations to the functions h_i computed using either direct linearization along the nominal trajectory, or by using a project-and-linearize technique if any h_i is a concave function [25]. Note that, if the state constraints h_i are all affine then $\mathcal{X}_{\mathcal{F}} = \mathcal{P}_{\bar{x}}$. The matrix $Q_{\max}$ is then computed as the maximum volume ellipsoid so that

$$\bar{x} \oplus \mathcal{E}_{Q_{\max}} \subset \mathcal{P}_{\bar{x}} \subseteq \mathcal{X}_{\mathcal{F}} \tag{37}$$

where $\oplus$ denotes the Minkowski sum of a vector and set, and so the term $\bar{x} \oplus \mathcal{E}_{Q_{\max}}$ is equivalent to the ellipsoid defined by $Q_{\max}$ centered at the vector $\bar{x}$. The assumption that $\bar{x}$ lies in the strict interior of $\mathcal{X}_{\mathcal{F}}$ ensures that $Q_{\max}$ describes a full n_x -dimensional ellipsoid.

Using techniques described in [21], we can write the condition (37) for any time $t \in \left[t_0, t_f\right]$ as the following optimization problem:

$$\begin{aligned} Q_{\max}^{1/2}(t) = \arg\max_{Z} \ & \log\det Z \\ \text{s.t. } & \|Z a_i(t)\|_2 + a_i^\top \bar{x}(t) \le b_i(t), \quad i = 1, \dots, m_x, \\ & 0 \prec Z \prec x_{\max} I_{n_x}. \end{aligned} \tag{38}$$

When $t = t_f$, a set of conditions can be added to constrain $\mathcal{E}_{Q_{max}} \subseteq \mathcal{X}_{\mathcal{F}}^f$. By assuming that the set $\mathcal{X}_{\mathcal{F}}^f$ can be described in the same way as that of $\mathcal{X}_{\mathcal{F}}$, this set of constraints is identical to those that are shown in (38), albeit calculated using the data corresponding to $\mathcal{X}_{\mathcal{F}}^f$. The matrices $R_{\max}$ can be computed analogously.‡

Recall that, a quadratic funnel is defined by the (possibly time-varying) ellipsoids $\mathcal{E}_Q$ and $\mathcal{E}_{K Q_y K^\top}$. Further, note that we derived the convex feasible subsets (defined by $Q_{\max}$ and $R_{\max}$) for the state and control inputs. A sufficient condition to ensure that a quadratic funnel is fully contained within the original feasible domain defined by $\mathcal{X}_{\mathcal{F}}$ and $\mathcal{U}_{\mathcal{F}}$ is

$$\mathcal{E}_Q \subseteq \mathcal{E}_{Q_{\max}} \quad \text{and} \quad \mathcal{E}_{K Q_y K^\top} \subseteq \mathcal{E}_{R_{\max}}. \tag{39}$$

‡See [16] for more details.

By using the definition of an ellipsoid (9), one can show that these conditions are equivalent to

$$Q \preceq Q_{\max} \quad \text{and} \quad K Q_y K^\top \preceq R_{\max} \tag{40}$$

The former is affine in the variable Q, whereas the latter is not jointly convex in the variables Q_y and K. By Theorem 4 and using a Schur complement, we can rewrite the second matrix inequality in (40) to be

$$(1 + s\,(1+\rho))^2 \;\; K Q K^\top \preceq R_{\max} \quad \Leftrightarrow \quad \begin{bmatrix} (1+s\,(1+\rho))^{-2}\, Q^{-1} & K^\top \\ K & R_{\max} \end{bmatrix} \succeq 0 \tag{41}$$

Pre- and post-multiplying by $\operatorname{diag}\{Q, I\}$ the feasibility condition for the control input becomes

$$\begin{bmatrix} (1+s\,(1+\rho))^{-2}\, Q & Y^\top \\ Y & R_{\max} \end{bmatrix} \succeq 0 \tag{42}$$

Together, the constraints $Q \preceq Q_{\max}$ and (42) ensure that a quadratic funnel is contained in the feasible region defined by $\mathcal{X}_{\mathcal{F}}$ and $\mathcal{U}_{\mathcal{F}}$.

We are now in a position to layout the performance guarantee of the presented vision-based controller synthesis.

Theorem 6. *Define the regions of the state space within which the sensing is reliable as described in (10). Then,if the feasible state space $\mathcal{X}_{\mathcal{F}}$ describing the quadratic funnel $\mathcal{F}$ lies within the safe set $\mathcal{X}_r(\bar{x}(t))$, then the perception error remains bounded and the closed loop trajectories defined by the maximum quadratic funnel $\mathcal{F}_{\max}$ lie within $\mathcal{X}_r(\bar{x}(t))$.*

Proof. If $\mathcal{X}_f \subseteq \mathcal{X}_r(\bar{x}(t))$, then by the invariance of the quadratic funnel $\mathcal{E}_Q \subseteq \mathcal{X}_r(\bar{x}(t))$. Moreover, by Lemma 3, for all $x(t) \in \mathcal{E}_Q$, the perception error remains bounded, i.e. $e(\eta(t)) \leq s(t)\|\eta(t)\| + l(t)$. □

D. The Quadratic Funnel Synthesis Problem

In this section, we combine the properties of the quadratic funnel and pose a single optimization problem. The quadratic funnel synthesis problem for the nonlinear system (8) with perception error is summarized in Problem 1.

Problem 1. (Quadratic Funnel Synthesis). Given a nominal trajectory $\{\bar{x}(t), \bar{u}(t)\}_{t=t_0}^{t_f}$ that satisfies the nonlinear dynamics (2), an appropriate definition of the system (8), find the matrix-valued functions of time $Q(t), Y(t), K(t), M(t), M_e(t)$ and scalars $\tau(t), \tau_e(t), \alpha(t)$ and $\lambda(t)$ that solve the following optimization problem, for $t \in [t_0, t_f]$.

$$\max_{\substack{Q(\cdot),Y(\cdot),K(\cdot),M(\cdot),M_e(\cdot),\\ \tau(\cdot),\tau_e(\cdot),\alpha(\cdot),\lambda(\cdot)}} \quad \log\det Q\,(t_0) \tag{43a}$$

$$\textbf{subject to} \quad 0 \preceq Q \preceq Q_{\max},\; 0 \leq \tau,\; 0 \leq \tau_e,\; 0 \leq \alpha,\; 0 \leq \lambda,\; M \in \mathcal{M}_{\phi,\Omega},\; M_e \in \mathcal{M}_{\widetilde{\phi},\mathcal{E}_{Q_e}} \tag{43b}$$

$$\begin{bmatrix} \mathbf{Q}A^\top + A\mathbf{Q} - \dot{\mathbf{Q}} + \mathbf{Q} + T_1 & E + \tau \mathbf{F}_{cl}^\top M_{12} & B & \tau \mathbf{F}_{cl}^\top M_{11}\widetilde{G} & 0 \\ E^\top + \tau M_{12}^\top \mathbf{F}_{cl} & \tau M_{22} & 0 & \tau M_{12}^\top \widetilde{G} & 0 \\ * & * & [& T_2 &] \end{bmatrix} \preceq 0, \tag{43c}$$

$$\begin{bmatrix} (1+s\,(1+\rho))^{-2}\, Q & Y^\top \\ Y & R_{\max} \end{bmatrix} \succeq 0 \tag{43d}$$

The matrices T_1 and T_2 are given by (29), and $\mathbf{F}_{cl} = F\mathbf{Q} + G\mathbf{Y}$.

A quadratic funnel obtained by solving Problem 1 can be used to generate a feasible trajectory in the following manner. Let $\bar{x}\,(t_0)$ be the initial condition of the reference state trajectory. For any initial condition $x\,(t_0)$ such that $x\,(t_0) - \bar{x}\,(t_0) \in \mathcal{E}_{Q(t_0)}$, the state and control vectors are defined as

$$u(t) = \bar{u}(t) + \mathbf{K}(t)\, y(t), \tag{44a}$$

$$x(t) = x\,(t_0) + \int_{t_0}^{t} f(x(\tau), u(\tau))\mathrm{d}\tau, \tag{44b}$$

and will satisfy $\eta = x - \bar{x} \in \mathcal{E}_Q$ and $\xi = u - \bar{u} = Ky \in \mathcal{E}_{KQ_yK^\top}$, for all $t \in [t_0, t_f]$. The feasibility with respect to the constraint sets $\mathcal{X}_{\mathcal{F}}$ and $\mathcal{U}_{\mathcal{F}}$ then follows. Notice that dynamic feasibility follows from (44) by construction, and the control u is guaranteed to be stabilizing for the original nonlinear dynamics.

III. The Solution Strategy

Problem 1 is a nonlinear, nonconvex optimization problem with matrix variables. The main difficulty in solving Problem 1 lies in the fact that it is not possible (to the best of our knowledge) to express (43c) as a linear DMI in the problem's variables. Moreover, the inclusion $M \in \mathcal{M}_{\phi,\Omega}$ and $M_e \in \mathcal{M}_{\widetilde{\phi},\mathcal{E}_{Q_e}}$ can at best be approximated. In this section we will provide a solution strategy to solve the optimization presented in Problem 1.

Since a direct solution of Problem 1 is not possible in general, we use the iterative method presented in [16]. This iterative method uses the following observation: if we fix the variables $\{Q(t), Y(t), K(t), \tau(t), \tau_e(t), \alpha(t), \lambda(t)\}$, then Problem 1 is convex in M and M_e provided we have an appropriate description of the convex cone $\mathcal{M}_{\phi,\Omega}$ and $\mathcal{M}_{\widetilde{\phi},\mathcal{E}_{Q_e}}$; conversely, for fixed M and M_e, Problem 1 is convex in the variables $\{Q(t), Y(t), K(t), \tau(t), \tau_e(t), \alpha(t), \lambda(t)\}$. A natural and common technique is to alternate between fixing one set of variables and solving for the other.

Finally, the nomenclature " γ -iteration" comes from the fact that the matrix

$$M_\gamma = \begin{bmatrix} \gamma^2 I & 0 \\ 0 & -I \end{bmatrix} \tag{45}$$

is a valid local multiplier matrix if γ and γ_e are local Lipschitz constants for the nonlinear functions ϕ over the set $\mathcal{M}_{\phi,\Omega}$ and $\widetilde{\phi}$ over the set $\mathcal{M}_{\phi,\mathcal{E}_{Q_e}}$, respectively. By solving for γ and γ_e, we may easily construct local multiplier matrices that can be used to obtain a new iterate for the variables $\{Q(t), Y(t), K(t), \tau(t), \tau_e(t), \alpha(t), \lambda(t)\}$. We now discuss a strategy to compute γ for an arbitrary nonlinear function and sets.

A. The M-Problem

The M-problem is the subproblem that is obtained by fixing the variables $\{Q(t), Y(t), K(t), \tau(t), \tau_e(t), \alpha(t), \lambda(t)\}$ in Problem 1. The quadratic inequality (25) under the assumption that $M = M_\gamma$ becomes

$$w^\top w \le \gamma^2 h^\top h \quad \Longleftrightarrow \quad \|w\|_2 \le \gamma \|h\|_2 \tag{46}$$

By introducing a matrix $\Delta \in \mathbb{R}^{n_w \times n_h}$, we can then write $w = \Delta h$ and $\|\Delta\|_2 \le \gamma$ in place of the nonlinear function ϕ in (16). The same procedure is applied for (27). We introduce matrix $\Delta_e \in \mathbb{R}^{n_u \times n_y}$, and replace the nonlinear function $\widetilde{\phi}$ in (16) with $w_e = \Delta_e e$ and $\|\Delta_e\|_2 \le \gamma_e$. We can rewrite the closed-loop formation (16) as

$$\dot{\eta} = (A + B\mathbf{K} + E\Delta(\, F + G\mathbf{K}\,))\, \eta + (E\Delta\widetilde{G} + B\Delta_e)e, \tag{47}$$

where

$$\|\Delta\|_2 \le \gamma, \quad \|\Delta_e\|_2 \le \gamma_e$$

To compute, or estimate, the value of γ and γ_e we follow the sampling procedure introduced in [16]. We would like to find the points $\eta \in \mathcal{E}_Q$ and $e \in \mathcal{E}_{Q_e}$ that corresponds to the largest value of $\|\Delta\|_2$ and $\|\Delta_e\|_2$, respectively. In our quest to find the most non-conservative local Lipschitz constants, we attempt to use the dependency of the perception error $e(\eta)$ on η during sampling. To this end, *at each discretized temporal point* t, we spatially sample a set of N_s states from the state-funnel $\mathcal{E}_{Q(t)}$ denoted by $\{\eta_s\}_{s=1}^{N_s}$, with s denoting the spatial iterator. For each η_s, the perception error $e(\eta_s)$ is then sampled from $\mathcal{E}_{Q_e}$.

At time t, given each η_s, we define the inner optimization problem

$$\begin{aligned} \Gamma(\eta_s) = \min_{\Delta,\, \Delta_e} \quad & \|\Delta\|_2 + \|\Delta_e\|_2 \\ \text{s.t.} \quad & \dot{\eta}_s - (A(t) + B(t)K(t))\, \eta_s = E\Delta(F + GK(t))\eta_s + (E\Delta\widetilde{G} + B(t)\Delta_e)\, e(\eta_s) \end{aligned} \tag{48}$$

Using

$$x_s = \bar{x}(t) + \eta_s, \quad y_s = \eta_s + e(\eta_s), \quad \xi_s = K(t) y_s, \quad \text{and} \quad u_s = \bar{u}(t) + \xi_s,$$

we can compute the followings at each temporal point t:

$$(A + BK)\,, \quad (F + GK), \quad \dot{\eta}_s = f\,(x_s, u_s) - f(\bar{x}, \bar{u})$$

This inner problem is the key component that allows us to be as non-conservative as possible in our search for local Lipschitz constants. We denote its solution as

$$\Gamma(\eta_s) \coloneqq \|\Delta^*(\eta_s)\|_2 + \|\Delta_e^*(\eta_s)\|_2 \tag{49}$$

where $\Delta^*(\eta_s)$ and $\Delta_e^*(\eta_s)$ represent the minimizers of (48), i.e., the "smallest" matrices satisfying the nonlinear equations of the dynamic model at each η_s. The local Lipschitz constants (γ, γ_e) are then computed at each temporal point t:

$$\gamma(t) = \|\Delta^*\left(\eta^*(t)\right)\|_2, \qquad \gamma_e(t) = \left\|\Delta_e^*\left(\eta^*(t)\right)\right\|_2 \tag{50}$$

where

$$\eta^*(t) = \arg \max_{\eta \in \{\eta_s\}_{s=1}^{N_s}} \Gamma(\eta) \tag{51}$$

The outer optimization problem (51) is selecting the sample η_s that produces the maximum value of the inner optimization problem (48) across the N_s samples.

This spatiotemporal sampling procedure has been observed to be accurate enough for practical problems with a reasonable number of points N_s (typically on the order of 10^3 or 10^4 for an entire trajectory). Note that because it is possible to solve N_s independent optimization problems, each of which is quite small, this sampling procedure can be done very quickly and is parallelizable.

B. The Q-Problem

We now describe the Q-problem, which is the subproblem that is obtained by fixing the local multiplier matrices M and M_e in Problem 1 and solving for the variables $\{Q(t), Y(t), \tau(t), \tau_e(t), \alpha(t), \lambda(t)\}$.

Theorem 7. *For fixed multiplier matrices M and M_e, the output controller $\xi = \mathbf{K}(t)y(t)$ satisfies the conditions in (23) if and only if there exists $Q \geq 0$, $Y \in \mathbb{R}^{n_u \times n_x}$ and scalars $\widetilde{\tau}, \widetilde{\tau}_e, \widetilde{\alpha}, \widetilde{\lambda} > 0$ such that for all $t \in [t_0, t_f]$,*

$$\begin{bmatrix} \mathbf{Q}A^\top + A\mathbf{Q} - \dot{\mathbf{Q}} + \mathbf{Q} + B\mathbf{Y} + \mathbf{Y}^\top B^\top & T_3 \\ * & T_4 \end{bmatrix} \leq 0, \qquad \gamma_e^2\,\widetilde{\boldsymbol{\lambda}} - \widetilde{\boldsymbol{\tau}}_{\boldsymbol{e}} \leq 0, \qquad R\widetilde{\boldsymbol{\alpha}} - \mu\widetilde{\boldsymbol{\lambda}} \leq 0 \tag{52}$$

where

$$\mathbf{F}_{cl} = F\mathbf{Q} + G\mathbf{Y}, \quad T_3 = \begin{bmatrix} \widetilde{\boldsymbol{\tau}}\,E & \widetilde{\boldsymbol{\tau}}_{\boldsymbol{e}}B & \gamma F_{cl}^\top & \mathbf{Q}C_\mu^\top & s\mathbf{Q} \end{bmatrix}, \quad T_4 = \boldsymbol{diag}\{-\widetilde{\boldsymbol{\tau}}I,\ -\widetilde{\boldsymbol{\tau}}_{\boldsymbol{e}}I,\ -\widetilde{\boldsymbol{\tau}}I,\ -\widetilde{\boldsymbol{\alpha}}I,\ -\widetilde{\boldsymbol{\lambda}}I\} \tag{53}$$

Proof. Consider the differential matrix inequality (43c). Due to the form of the local multiplier matrix M_γ in (45), we know that $M_{11} \geq 0$. In this case, the matrix inequality (43c) is equivalent to $\boldsymbol{\lambda}R - \boldsymbol{\alpha}\mu \leq 0$ and the following

$$\begin{bmatrix} \mathbf{Q}A^\top(t) + A(t)\mathbf{Q} - \dot{\mathbf{Q}} + \mathbf{Q} \\ \quad +B(t)\mathbf{Y} + \mathbf{Y}^\top B^\top(t) & \mathcal{T}_1 \\ * & \mathcal{T}_2 \end{bmatrix} + \begin{bmatrix} \gamma\mathbf{F}_{cl}^\top \\ 0 \\ 0 \\ \gamma\widetilde{G}^\top \end{bmatrix} (\tau I) \begin{bmatrix} \gamma\mathbf{F}_{cl} & 0 & 0 & \gamma\widetilde{G} \end{bmatrix} + \begin{bmatrix} \mathbf{Q}C_\mu^\top \\ 0_{3\times1} \end{bmatrix} (\alpha I) \begin{bmatrix} C_\mu\mathbf{Q} & 0_{1\times3} \end{bmatrix} + \begin{bmatrix} s\mathbf{Q} \\ 0_{3\times1} \end{bmatrix} (\lambda I) \begin{bmatrix} s\mathbf{Q} & 0_{1\times3} \end{bmatrix} \leq 0$$

where

$$\mathcal{T}_1 = \begin{bmatrix} E & B & 0 \end{bmatrix}, \qquad \mathcal{T}_2 = \mathbf{diag}\{-\tau I,\ -\tau_e I,\ \gamma_e^2\tau_e I - \lambda I\}$$

If E and $B \neq 0$ then we must have $\tau, \tau_e, \lambda, \alpha > 0$. Hence we can use a Schur complement to arrive at the equivalent condition

$$\begin{bmatrix} \mathbf{Q}A^\top(t) + A(t)\mathbf{Q} - \dot{\mathbf{Q}} + \mathbf{Q} \\ \quad +B(t)\mathbf{Y} + \mathbf{Y}^\top B^\top(t) & \mathcal{T}_3 \\ * & \mathcal{T}_4 \end{bmatrix} \leq 0, \qquad \gamma_e^2\tau_e I - \lambda I \leq 0$$

where

$$\mathcal{T}_3 = \begin{bmatrix} E & B & \gamma\mathbf{F}_{cl}^\top & \mathbf{Q}C_\mu^\top & s\mathbf{Q} \end{bmatrix}, \qquad \mathcal{T}_4 = \mathbf{diag}\{-\tau I,\ -\tau_e I,\ -\tau^{-1}I,\ -\alpha^{-1}I,\ -\lambda^{-1}I\}$$

Let $\widetilde{\tau} \coloneqq \tau^{-1}$, $\widetilde{\tau}_e \coloneqq \tau_e^{-1}$, $\widetilde{\alpha} \coloneqq \alpha^{-1}$ and $\widetilde{\lambda} \coloneqq \lambda^{-1}$. Pre- and post-multiplying the above matrix-inequality by $\mathbf{diag}\{I,\ \widetilde{\tau},\ \widetilde{\tau}_e,\ I,\ I,\ I\}$ gives the proof.

□

The continuous-time Q-problem is summarized below in Problem 2. It is assumed that a nominal trajectory and system definition are all given.

Problem 2. (Q-Problem). Given local multiplier matrices $M \in \mathcal{M}_{\phi,\Omega}$ and $M_e \in \mathcal{M}_{\widetilde{\phi},\mathcal{E}_{Q_e}}$, find the matrix-valued functions of time $Q(t)$, $Y(t)$ and scalars $\widetilde{\tau}(t), \widetilde{\tau}_e(t), \widetilde{\alpha}(t), \widetilde{\lambda}(t)$ that solve the following optimization problem

$$\max_{\substack{\widetilde{\tau}(t),\widetilde{\tau}_e(t),\widetilde{\alpha}(t),\widetilde{\lambda}(t),\\ Q(\cdot),Y(\cdot)}} \quad \log\det Q\,(t_0) \tag{54a}$$

$$\textbf{subject to} \quad 0 \le Q \le Q_{\max},\ 0 \le \widetilde{\tau}(t),\ 0 \le \widetilde{\tau}_e(t),\ 0 \le \widetilde{\alpha}(t),\ 0 \le \widetilde{\lambda}(t), \tag{54b}$$

$$\gamma_e^2\,\widetilde{\boldsymbol{\lambda}} - \widetilde{\boldsymbol{\tau}}_{\boldsymbol{e}} \le 0, \quad R\widetilde{\boldsymbol{\alpha}} - \mu\widetilde{\boldsymbol{\lambda}} \le 0 \tag{54c}$$

$$\begin{bmatrix} \mathbf{Q}A^\top + A\mathbf{Q} - \dot{\mathbf{Q}} + \mathbf{Q} + B\mathbf{Y} + \mathbf{Y}^\top B^\top & T_3 \\ * & T_4 \end{bmatrix} \le 0 \tag{54d}$$

$$\begin{bmatrix} (1 + s\,(1+\rho))^{-2}\,\mathbf{Q} & \mathbf{Y}^\top \\ \mathbf{Y} & R_{\max} \end{bmatrix} \ge 0 \tag{54e}$$

where T_3 and T_4 are defined in (53).

In order to solve the Q-problem numerically, we make the following assumption regarding the continuous-time problem data.

Assumption 8. *Let $M(t)$ denote any of the matrices $Q(t), Q_{\max}(t), Y(t), R_{\max}(t), A(t)$ or $B(t)$. We assume that $M(t)$ can be expressed or approximated as the convex combination*

$$M(t) = \sum_{i=1}^{n_M} \sigma_i(t) M_i \tag{55}$$

for some integer $n_M > 1$, constant matrices M_i, and interpolating functions $\sigma_i(t) \ge 0$ such that $\sum_{i=1}^{n_M} \sigma_i(t) = 1$.

By using the *temporal matrix decomposition methods* discussed in [16], the DMI in (54d) can be written as a set of $\frac{1}{2}n_M\,(n_M + 1)$ LMIs. These LMIs constitute a set of sufficient conditions for the satisfaction of the DMI (54d). The constraints (54d) can be written as a set of LMIs by using (55) and constraining each sum and in the resulting expression individually.

C. Algorithm Summary

Having defined the $Q-$ and $M-$problems independently, we now collect them together in order to design a convergent algorithm that is capable of solving problem 1. As it is discussed in [16], the premise of the $\gamma-$iteration is that we can decrease the upper bound, $Q_{\max}$, until we are able to synthesize a funnel $Q \approx Q_{\max}$. We measure how close a given solution of the $Q-$problem is to achieving the upper bound of $Q \approx Q_{\max}$ by using the fill ratio, defined by

$$\kappa = \min_{i=1,\ldots,n_x} \left(\frac{\mathrm{proj}_i\, \mathcal{E}_Q}{\mathrm{proj}_i\, \mathcal{E}_{Q_{\max}}} \right)^{1/2} \tag{56}$$

Algorithm 1 describes the γ-iteration used to solve the quadratic funnel synthesis problem. For the proof of convergence, see [16].

IV. RCVS Control for target Tracking Problems

In this section, we present a motivating setup for vision-based navigation and control of a spacecraft where we can showcase the performance of the perception error modeling and the robust controller synthesis developed in Section II. For that, consider a target tracking problem, where the target spacecraft (so-called the target) is uncontrolled and possibly uncooperative. Thus, by using the image sensors available on the ego spacecraft (so-called the ego), we adopt a computer vision approach to predict target's relative position. We used a high-fidelity astrodynamic simulator [26] along with a photo-realistic simulator [27] to replicate the actual system. This environment emulates the space conditions in LEO. A

Algorithm 1 The γ-iteration designed to solve the quadratic funnel synthesis problem.

Require: A nominal trajectory $\{\bar{x}(t), \bar{u}(t)\}_{t=t_0}^{t_f}$, the system matrices for (16), a tolerance $\kappa_{tol} \in (0, 1)$.

solve (38) for $Q_{\max}$ and $R_{\max}$
set $M_\gamma \leftarrow 0$ and $M_{\gamma_e} \leftarrow 0$
solve the Q-problem to get $\{Q^{(0)}, Y^{(0)}\}$ ▷ Problem 2
update $Q_{\max}^{(0)} \leftarrow Q^{(0)}$ and $Y_{\max}^{(0)} \leftarrow Y^{(0)}$
for $k = 1, \cdots$ **do**
    solve the M-problem to get $M_\gamma^{(k)}$ and $M_{\gamma_e}^{(k)}$ using $\{Q_{\max}^{(k-1)}, Y_{\max}^{(k-1)}\}$ ▷ see (51)
    solve the Q-problem to get $\{Q^{(k)}, Y^{(k)}\}$ using $M_\gamma^{(k)}$ and $M_{\gamma_e}^{(k)}$ ▷ Problem 2
    if $\kappa^{(k)}(t_0) \geq \kappa_{tol}$ **then** ▷ see (56)
        Converged.
        break
    else
        $Q_{\max}^{(k)} \leftarrow \zeta Q_{\max}^{(k-1)}$ and $Y_{\max}^{(k)} \leftarrow \zeta Y_{\max}^{(k-1)}$ ▷ contraction step, see [16]
    end if
end for

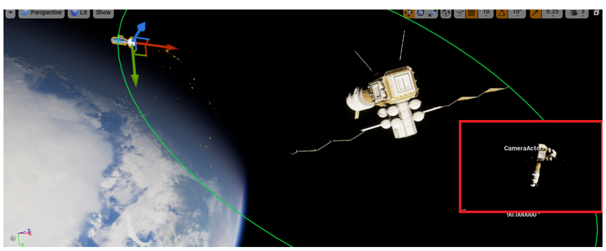


Fig. 3 A snapshot of the spacecraft simulation setup showing the ego (denoted on the left with its attached body frame) orbiting around an uncontrolled satellite target (the white object in the middle) on a fix flyby orbit (denoted by a green circle). The in-picture on the right shows the target in the ego's camera frame.

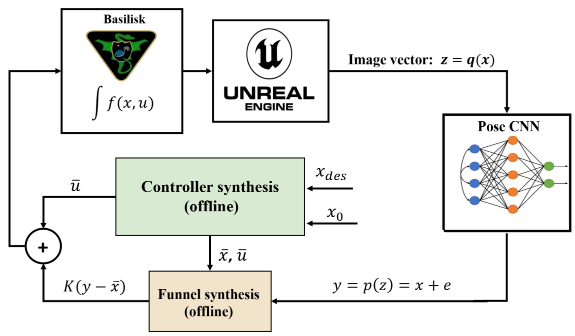


Fig. 4 The simulation setup developed for emulating the real-world settings for applications of the proposed machinery. The astrodynamics simulator [26] and the developed visual simulator are replicating the actual system with possibility of generating high-quality image data. The state estimated from the image data using the CNN module is then fed into the RCVS controller module for robust feedback signal synthesis.

sample snapshot of the environment is illustrated in Figure 3, and the developed pipeline is showcased in Figure 4. The details of the CNN-based perception module and the overall space flight simulation setup can be found in [14].

Assuming that the ego is inserted in a passive elliptical orbit around the target, the mission starts when the ego's camera points away from the target's center of mass but has the target in its field of view. The goal is to drive the ego's attitude and angular rates such that the target's center of mass is placed at the center of the ego's camera frame. Thus, herein, we lay out the attitude dynamics of the ego using a modified Rodriguez parameters approach.

To obtain ego's attitude dynamics in the Earth fixed frame ($\mathcal{F}_\mathcal{E}$), we denote the positive definite inertial tensor by $J \in \mathbb{R}^{3\times3}$ and the angular velocity by $\boldsymbol{\omega}^E \in \mathbb{R}^3$. The torque input is denoted by $\boldsymbol{u} \in \mathbb{R}^3$ and thus, the rigid-body dynamics of the ego in the Earth fixed frame $\mathcal{F}_\mathcal{E}$, are given by:

$$J\dot{\boldsymbol{\omega}}^E = \left(J\boldsymbol{\omega}^E\right) \times \boldsymbol{\omega}^E + \boldsymbol{u}^E + \boldsymbol{d}_{\text{ext}}\,,$$

where $\boldsymbol{d}_{\text{ext}}$ represents the external torque acting on the ego, which is assumed to be negligible in this setting for brevity. Next, by using Modified Rodriguez parameters (MRP), $\boldsymbol{q}^E = [q_1 \;\; q_2 \;\; q_3]^\top \in \mathbb{R}^3$ with $\boldsymbol{q}^E = \boldsymbol{e}.\tan\frac{\phi}{2}$ (such that $\boldsymbol{e}$ is the unit vector denoting the axis of rotation and ϕ is the rotation angle), the attitude dynamics can be described as follows:

$$\frac{d}{dt}\begin{bmatrix} \boldsymbol{q}^E \\ \boldsymbol{\omega}^E \end{bmatrix} = f(\boldsymbol{q}^E, \boldsymbol{\omega}^E, \boldsymbol{u}) = \begin{bmatrix} \frac{1}{2}\left(\mathbf{I}\left(\frac{1-(\boldsymbol{q}^E)^T\boldsymbol{q}^E}{2}\right) + \boldsymbol{q}^E\,(\boldsymbol{q}^E)^T + S(\boldsymbol{q}^E)\right)\boldsymbol{\omega}^E \\ J^{-1}\left((J\boldsymbol{\omega}^E)\times\boldsymbol{\omega}^E + \boldsymbol{u}\right) \end{bmatrix} \tag{57}$$

where $S(\boldsymbol{q}^E) := \begin{bmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{bmatrix}$. Note that target pose estimation depends on both the relative orbital motion between ego and target, and the attitude dynamics of ego. The relative orbital motion follows the Hill-Clohessy-Wiltshire equations [28] that can be solved for an exact solution. We denote the ego's position with respect to the target's body frame of reference, by $r = [r_x,\ r_y,\ r_z]^\top \in \mathbb{R}^3$. Following the Hill-Clohessy-Wiltshire equations in phase magnitude form we have

$$\ddot{r}_x = \rho_x \sin(nt + a_x)\,, \quad \ddot{r}_y = \rho_y + 2\rho_x \cos(nt + a_x)\,, \quad \ddot{r}_z = \rho_z \sin(nt + a_x) \tag{58}$$

where $n = \sqrt{\mu/a^3}$ and a refers to the orbital radius of the target and μ is the standard gravitational parameter. In (58), (a_x, a_y, a_z) denote the semi major axes and (ρ_x, ρ_y, ρ_z) give the phase magnitude of the passive relative elliptical orbit. The formulation of phase magnitude and selection of initial conditions for the ego spacecraft in the reltive elliptical orbit have been detailed in [29]

For this problem, denote the relative states between target and ego spacecraft as $x(t) := \left[r,\ \dot{r},\ q^E,\ w^E\right]^\top \in \mathbb{R}^{12}$. We disregard the target's orientation, as it is not needed for the current problem. The relative dynamics are described using (58) and desired attitude is calculated using difference between relative target position and ego camera axis. This desired attitude is used as reference state for the attitude dynamics defined by (57). Thus the task of the observation module is to detect the target and determine its position in the ego's camera frame. The camera is fixed on the ego's body frame facing its principal x-axis. For simplicity of presentation, we assume the ego's body frame and its camera frame are identical.

Along the relative dynamics, we consider a sensory observation $z(t) \in \mathbb{R}^{m\times m}$ as

$$z(t) = Q(x(t)) \tag{59}$$

that represents the images taken at relative state $x(t)$. $Q : \mathbb{R}^{12} \to \mathbb{R}^{m\times m}$ is a nonlinear and potentially quite high dimensional mapping. The CNN-based pose estimation techniques are then adopted to represent a *perception map* $p : \mathbb{R}^{m\times m} \to \mathbb{R}^3$ that imperfectly recovers the relative position $x(t)$ in the camera frame; that is,

$$p(z(t)) = C\,x(t) + e_p(x(t)) \tag{60}$$

where e_p denotes the perception error that is dependent on the relative states. To model the perception map and study its error characteristics, as discussed in section II.B, we design a nominal nonlinear controller for given initial relative states $x(0)$. Such nominal controller generates a nominal trajectory $\{\overline{x}(t), \overline{u}(t)\}_{t=t_0}^{t_f}$, depicted in Figure 5, as seen in the ego's camera frame. We then collect the initial training dataset $\mathcal{S}_0 = \{(x_d, z_d)\}_{d=1}^{N_0}$ from the nominal trajectory and train the neural network.

Based on the assumption that the ego is in a fixed flyby orbit around the target, we focus our efforts on designing the torque vector $\mathbf{u}$ that tracks the target position in the ego's camera frame. Hence, starting off from the nominal trajectory with error, we define

$$\eta(t) := C\;x(t) - C\;\overline{x}(t) = \begin{bmatrix} q^E(t) \\ w^E(t) \end{bmatrix} - \begin{bmatrix} \overline{q^E}(t) \\ \overline{w^E}(t) \end{bmatrix}$$

as the tracking error and $\xi = u - \bar{u}$ as the correction signal. Figure 6 showcases the error around the nominal trajectory with respect to time and η.

The matrices A and B are the partial derivatives of the attitude dynamics in (57) along the nominal trajectory, and the parameters F, G and E are constructed using a total of four ($n_h = 4$) nonlinear channels to be

$$F = \begin{bmatrix} I_3 & 0_{3\times3} \\ I_3 & I_3 \\ I_3 & I_3 \\ I_3 & I_3 \end{bmatrix}, \qquad G = 0_{12\times6}, \qquad E = \begin{bmatrix} I_3 & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & I_3 & I_3 & I_3 \end{bmatrix}.$$

See [14] for further details regarding the neural network setup.

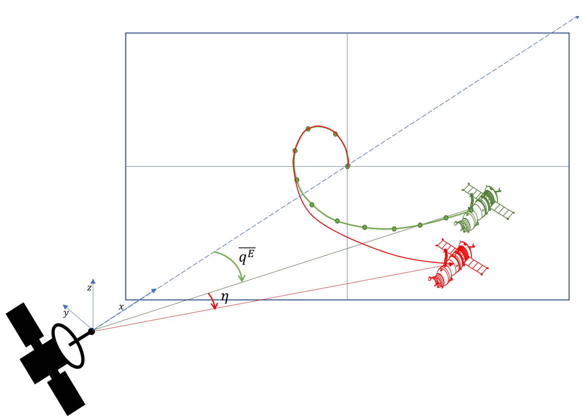


Fig. 5 Nominal trajectory in green as seen in the camera frame. For some initial condition which deviate form the nominal as shown in red, we use the robust controller such that the error between the new state and nominal given by η is driven to zero.

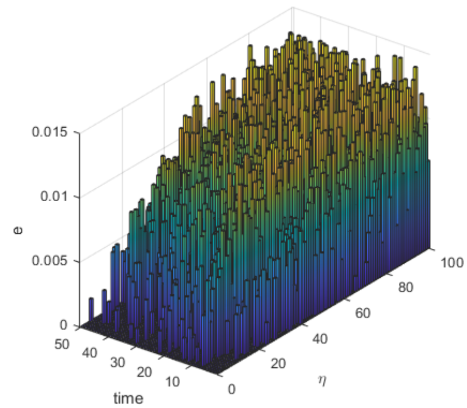


Fig. 6 The error profile in the neighborhood of the nominal trajectory, for $T = 50$ seconds given 100 deviation samples per time stamp.

Note that for this problem $n_x = 6$, $n_u = 3$ and $J = \textbf{diag}\{900,\ 900,\ 900\}\quad kg\,m^2$. The quadratic funnel synthesis problem in section II.D, provides a robust controller that renders a UUB closed-loop system.

Figure 7 depicts the computed quadratic funnel. In Figure 7a, the ellipsoid $\mathcal{E}_Q$ is projected onto each state dimension and depicted as the shaded grey area. The red trajectories correspond to test cases for which an initial condition was randomly (uniformly) selected from the ellipsoid $\mathcal{E}_Q(t_0)$, and the nominal control and correction law were used to numerically integrate the equations of motion according to (44). Figure 7b shows the ellipsoid $\mathcal{E}_{R_{\max}}$ projected into each control dimension along with the corresponding control trajectories from each test case (where the first few iterations are enlarged in Section IV). Finally, the code to reproduce these results is available here [30].

V. Conclusions and Future Directions

We have developed a robust controller synthesis method for Vision-based Spacecraft guidance and control. The proposed approach synthesizes a feedback signal directly from image data. Therein, we adopted a CNN as a perception module and characterized the associated error along a nominal trajectory defined by a nonlinear controller. This nominal trajectory is mission-specific, and accordingly, we developed a robust control synthesis procedure —building on the invariant funnel concept. We have used the guaranteed state-dependent error bounds to synthesize a feedback signal directly from the image data and thus close the loop with an image sensor– robustly. Finally, we have developed a simulation environment using an astrodynamics simulator and a photo-realistic simulator to study the performance of the proposed machinery on a target spacecraft tracking problem.

The future directions include expanding the proposed approach for multi-spacecraft formation control and reconfiguration applications, as well as, data-driven dynamics modeling and control.

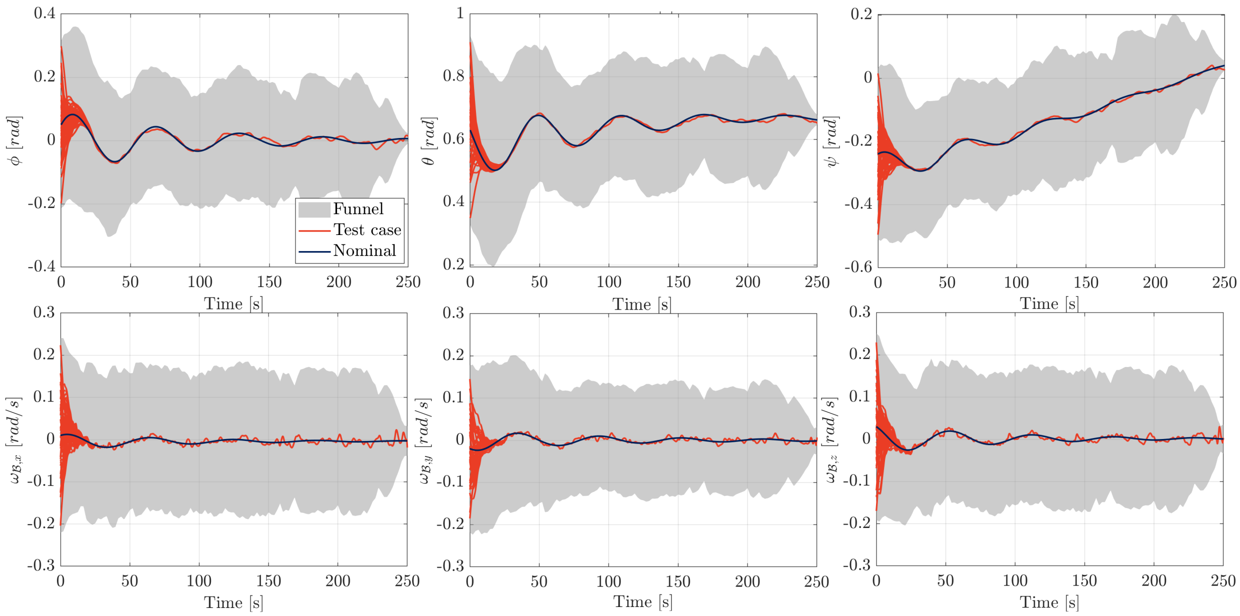
Acknowledgments

Part of this research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

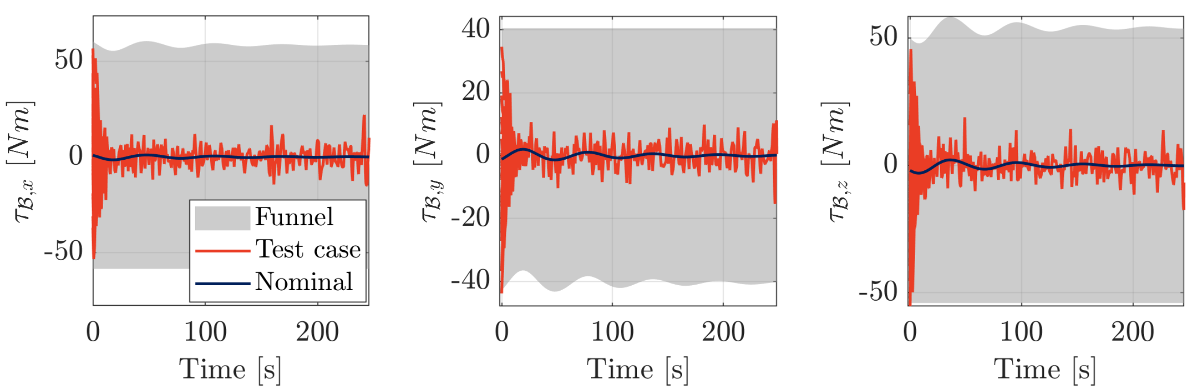
References

[1] Stoll, E., Letschnik, J., Walter, U., Artigas, J., Kremer, P., Preusche, C., and Hirzinger, G., "On-orbit servicing," *IEEE Robotics & Automation Magazine*, Vol. 16, No. 4, 2009, pp. 29–33. URL `https://elib.dlr.de/69286/`.

[2] Bonnal, C., Ruault, J.-M., and Desjean, M.-C., "Active debris removal: Recent progress and current trends," *Acta Astronautica*,



(a) In state space $\mathcal{E}_Q$



(b) In control space $\mathcal{E}_{R_{\max}}$

Fig. 7 The quadratic funnel computed by the γ-iteration for the target tracking problem. The initial condition of each test case was randomly sampled from the funnel entry.

Vol. 85, 2013, pp. 51–60. https://doi.org/https://doi.org/10.1016/j.actaastro.2012.11.009, URL https://www.sciencedirect.com/science/article/pii/S0094576512004602.

[3] Ellery, A., Kreisel, J., and Sommer, B., "The case for robotic on-orbit servicing of spacecraft: Spacecraft reliability is a myth," *Acta Astronautica*, Vol. 63, No. 5, 2008, pp. 632–648. https://doi.org/https://doi.org/10.1016/j.actaastro.2008.01.042, URL https://www.sciencedirect.com/science/article/pii/S0094576508001197.

[4] Lillie, C. F., "On-orbit assembly and servicing of future space observatories," *Space Telescopes and Instrumentation I: Optical, Infrared, and Millimeter*, Vol. 6265, edited by J. C. Mather, H. A. MacEwen, and M. W. M. de Graauw, International Society for Optics and Photonics, SPIE, 2006, pp. 767 – 778. https://doi.org/10.1117/12.672528, URL https://doi.org/10.1117/12.672528.

[5] Feng, Y., Zhen, H., Bing, Q., and Xiaoting, Y., "Stereo-vision-based relative pose estimation for the rendezvous and docking of non-cooperative Satellites," *Mathematical Problems in Engineering*, 2014. https://doi.org/10.1155/2014/461283.

[6] Song, L., Li, Z., and Ma, X., "Autonomous rendezvous and docking of an unknown tumbling space target with a monocular camera," *Proceedings of 2014 IEEE Chinese Guidance, Navigation and Control Conference*, 2014, pp. 1008–1013. https://doi.org/10.1109/CGNCC.2014.7007346.

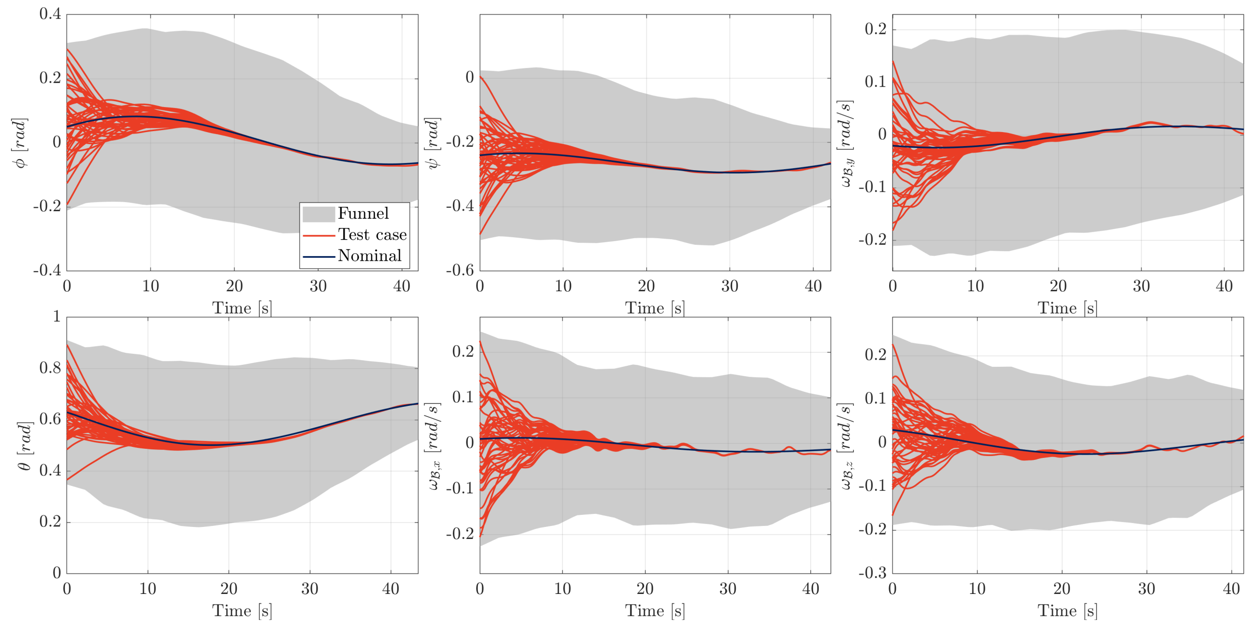


Fig. 8 The first few iteration of the same quadratic funnel for the target tracking problem in Figure 7a.

[7] Cassinis, L. P., Fonod, R., Gill, E., Ahrns, I., and Fernandez, J. G., "CNN-based Pose estimation system for close-proximity operations around uncooperative spacecraft," *AIAA Scitech 2020 Forum*, 2020, p. 1. https://doi.org/10.2514/6.2020-1457, URL https://arc.aiaa.org/doi/abs/10.2514/6.2020-1457.

[8] Sharma, S., Beierle, C., and D'Amico, S., "Pose estimation for non-cooperative spacecraft rendezvous using convolutional neural networks," *2018 IEEE Aerospace Conference*, 2018, pp. 1–12. https://doi.org/10.1109/AERO.2018.8396425.

[9] Harvard, A., Capuano, V., Shao, E. Y., and Chung, S.-J., "Spacecraft pose estimation from monocular images using neural network based keypoints and visibility maps," *AIAA Scitech 2020 Forum*, 2020, p. 1. https://doi.org/10.2514/6.2020-1874, URL https://arc.aiaa.org/doi/abs/10.2514/6.2020-1874.

[10] Sonawani, S. D., Alimo, R., Detry, R., Jeong, D., Hess, A., and Amor, H. B., "Assistive relative pose estimation for on-orbit assembly using Convolutional Neural Networks," *CoRR*, Vol. abs/2001.10673, 2020. URL https://arxiv.org/abs/2001.10673.

[11] Proenca, P. F., and Gao, Y., "Deep learning for spacecraft pose estimation from photorealistic rendering," *Proceedings - IEEE International Conference on Robotics and Automation*, 2020, pp. 6007–6013. https://doi.org/10.1109/ICRA40945.2020.9197244, URL https://arxiv.org/pdf/1907.04298.

[12] Fuentes-Pacheco, J., Ruiz-Ascencio, J., and Rendón-Mancha, J. M., "Visual simultaneous localization and mapping: a survey," *Artificial Intelligence Review*, Vol. 43, No. 1, 2012, pp. 55–81. https://doi.org/10.1007/s10462-012-9365-8.

[13] Alhwarin, F., Ferrein, A., and Scholl, I., "IR stereo kinect: Improving depth images by combining structured light with IR stereo," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Vol. 8862, 2014, pp. 409–421. https://doi.org/10.1007/978-3-319-13560-1.

[14] Becktor, J., Seto, W., Deole, A., Bandyopadhyay, S., Rahimi, N., Talebi, s., Mesbahi, M., and Rahmani, A., "Robust vision-based multi-spacecraft guidance navigation control using CNN-based pose estimation," *2022 IEEE Aerospace Conference*, IEEE, 2022, pp. –.

[15] Dean, S., Matni, N., Recht, B., and Ye, V., "Robust guarantees for perception-based control," *Proceedings of Machine Learning Research*, 2019, pp. 1–21. URL http://arxiv.org/abs/1907.03680.

[16] Reynolds, T. P., Malyuta, D., Mesbahi, M., Açıkmeşe, B., and Carson, J. M., "Funnel synthesis for the 6-dof powered descent guidance problem," *AIAA Scitech 2021 Forum*, 2021, pp. 1–21. https://doi.org/10.2514/6.2021-0504.

[17] Khalil, H. K., *Nonlinear Systems*, Vol. 3, Upper Saddle River, NJ: Prentice hall, 2002.

[18] Talebi, S., Alemzadeh, S., Rahimi, N., and Mesbahi, M., "On regularizability and its application to online control of unstable LTI systems," *IEEE Transactions on Automatic Control*, 2021, pp. 1–1. https://doi.org/10.1109/TAC.2021.3131148.

[19] Rosenfeld, J. A., and Kamalapurkar, R., "Dynamic mode decomposition with control liouville operators," *arXiv preprint arXiv:2101.02620*, 2021.

[20] Reynolds, T. P., "Computational Guidance and Control for Aerospace Systems," Ph.D. thesis, University of Washington, 2020. URL http://hdl.handle.net/1773/46722.

[21] S. Boyd, E. F., L. El Ghaoui, and Balakrishnan, V., *Linear Matrix Inequalities in System and Control Theory*, Vol. 3, Studies in Applied Mathematics, SIAM, Philadelphia, PA, 1994.

[22] Açıkmeşe, B., Carson III, J. M., and Bayard, D. S., "A robust model predictive control algorithm for incrementally conic uncertain/nonlinear systems," *International Journal of Robust and Nonlinear Control*, Vol. 21, No. 5, 2011, pp. 563–590. https://doi.org/https://doi.org/10.1002/rnc.1613, URL https://onlinelibrary.wiley.com/doi/abs/10.1002/rnc.1613.

[23] Açikmese, B., and Corless, M. J., "Stability analysis with quadratic Lyapunov functions: Some necessary and sufficient multiplier conditions," *Syst. Control. Lett.*, Vol. 57, 2008, pp. 78–94.

[24] D'Alto, L., and Corless, M., "Incremental quadratic stability," *Numerical Algebra, Control & Optimization*, Vol. 3, No. 1, 2013, p. 175.

[25] Mao, Y., Dueri, D., Szmuk, M., and Açıkmeşe, B., "Successive convexification of non-Convex optimal control problems with state constraints," *IFAC-PapersOnLine*, Vol. 50, No. 1, 2017, pp. 4063–4069. https://doi.org/10.1016/j.ifacol.2017.08.789.

[26] Kenneally, P. W., Piggott, S., and Schaub, H., "Basilisk: A flexible, scalable and modular astrodynamics simulation framework," *Journal of Aerospace Information Systems*, Vol. 17, No. 9, 2020, pp. 496–507. https://doi.org/10.2514/1.I010762, URL https://doi.org/10.2514/1.I010762.

[27] Epic Games, "Unreal Engine," , 2020. URL https://www.unrealengine.com.

[28] Clohessy, W., and Wiltshire, R., "Terminal guidance system for satellite rendezvous," *Journal of the Aerospace Sciences*, Vol. 27, No. 9, 1960, pp. 653–658. https://doi.org/10.2514/8.8704.

[29] Alfriend, K. T., Vadali, S. R., Gurfil, P., How, J. P., and Breger, L. S., *Chapter 5 - Linear Equations of Relative Motion*, Butterworth-Heinemann, Oxford, 2010. https://doi.org/https://doi.org/10.1016/B978-0-7506-8533-7.00210-4, URL https://www.sciencedirect.com/science/article/pii/B9780750685337002104.

[30] Rahimi, N., Talebi, S., Deole, A., Mesbahi, M., Bandyopadhyay, S., and Rahmani, A., "Code: Robust controller synthesis for vision-based spacecraft guidance and control," , 2021. URL https://github.com/NiyoushaRahimi/Robust-Controller-Synthesis-forVision-based-Spacecraft-Guidance-and-Control.