Satellite Rendezvous and Conjunction Avoidance: Case Studies in Verification of Nonlinear Hybrid Systems*

Taylor T. Johnson[1], Jeremy Green[1], Sayan Mitra[1], Rachel Dudley[2], and R. Scott Erwin[3]

[1] University of Illinois at Urbana-Champaign
Urbana, IL 61801, USA
{johnso99, jdgreen4, mitras} @illinois.edu
[2] Iowa State University
Ames, IA 50011, USA
rfdudley@iastate.edu
[3] Air Force Research Laboratory
Albuquerque, NM 87116, USA

Abstract. Satellite systems are beginning to incorporate complex autonomous operations, which calls for rigorous reliability assurances. Human operators usually plan satellite maneuvers in detail, but autonomous operation will require software to make decisions using noisy sensor data and problem solutions with numerical inaccuracies. For such systems, formal verification guarantees are particularly attractive. This paper presents automatic verification techniques for providing assurances in satellite maneuvers. The specific reliability criteria studied are rendezvous and conjunction avoidance for two satellites performing orbital transfers. Three factors pose challenges for verifying satellite systems: (a) incommensurate orbits, (b) uncertainty of orbital parameters after thrusting, and (c) nonlinear dynamics. Three abstractions are proposed for contending with these challenges: (a) quotienting of the state-space based on periodicity of the orbital dynamics, (b) aggregation of similar transfer orbits, and (c) overapproximation of nonlinear dynamics using hybridization. The method's feasibility is established via experiments with a prototype tool that computes the abstractions and uses existing hybrid systems model checkers.

1 Introduction

As greater numbers of satellites are deployed and maintained in space, there is a growing need for autonomy in their operation. Software-based control systems enable autonomy by performing routine tasks automatically and minimize the need for human supervision. Given the high cost of space systems, a high level of reliability assurance is crucial. To provide such assurances, formal methods can complement traditional testing and simulation-based methods, and can also help find defects early in the design process.

* Most of this research was conducted under the Air Force's 2011 Summer Faculty Fellowship Program and Space Scholars Program at the Air Force Research Laboratory at Kirtland Air Force Base. The Illinois researchers were also supported by NSF CAREER Grant 1054247.

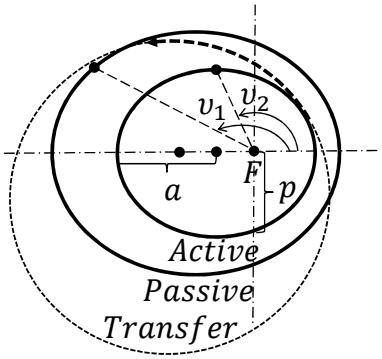

Fig. 1. Orbital transfer for two satellites: ν_1 and ν_2 are the angular positions of the passive and active satellite, respectively, a is the *semi-major axis* (max distance from the ellipse center to the ellipse edge), and p is the *semi-latus rectum* (distance from foci F to ellipse in direction perpendicular to the semi-major axis).

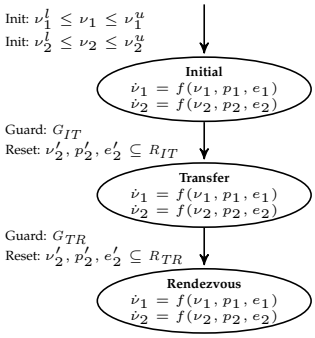

Fig. 2. Hybrid automaton for a two-stage rendezvous maneuver. The angular positions of the passive (ν_1) and active (ν_2) satellites evolve according to the nonlinear dynamics $\dot{\nu}_i = f(\nu_i, p_i, e_i) = \sqrt{\mu/p_i^3}(1 + e_i \cos \nu_i)^2$. Initial conditions are nondeterministically selected from the indicated ranges.

In this paper, we propose and validate a methodology for verifying autonomous operations between a pair of satellites. To the best of our knowledge, this is the first application of automatic verification to autonomously maneuvering satellite systems. The sound overapproximation approach presented in this paper allows us to nondeterministically model inaccuracies due to sensor measurements and numerical errors, which can cause serious errors in simulations. A *passive satellite* moves in a specific orbit, and an *active satellite* performs a software-controlled orbital transfer (see Fig. 1). Orbital transfers are performed when, for example, one satellite services (refuels or repairs) another satellite [9]. We aim to verify two properties: (A) **conjunction avoidance**: two passive (non-thrusting) satellites do not come closer than a certain distance, and (B) **rendezvous**: given a passive and an active satellite, the two satellites come closer than a certain distance of each other during a specified interval of time.

Our approach for verification is first to compute the reach set of an abstraction of the system and then to check that this set satisfies the above properties. Consider two satellites on different orbits with periods T_1 and T_2. The state of the satellites on their orbits is completely specified by the angular positions ν_1 and ν_2. In verifying rendezvous or conjunction avoidance, we are interested in computing the set of angular position pairs (ν_1, ν_2) that are reachable from a given set of initial angular positions. However, we have to overcome the following technical challenges in computing the reach set.

First, we observe that for *incommensurate orbits* (orbits with an irrational ratio of periods T_1/T_2) the unbounded-time reach set is dense in the set of all possible relative angular positions, $[0, 2\pi]^2$. This means that for incommensurate orbits, all (ν_1, ν_2) pairs are eventually visited arbitrarily closely. Therefore, we will focus on bounded-time versions of rendezvous or conjunction avoidance. In conjunction avoidance, for example, it suffices to verify safety up to

a certain time horizon because new ground-based measurements are available that can be used as updated initial conditions.

Second, for the active satellite 2 to rendezvous with the passive satellite 1, 2 must burn its thrusters to enter a new orbit called a *transfer orbit* to intercept 1 (see Fig. 1). The transfer orbit 2 follows depends crucially on the position where it burns its thrusters. The magnitude and direction of the thrusting are determined by numerically solving a standard orbital dynamics problem called *Lambert's problem*. Due to such numerical methods and other sources of inaccuracy like sensor noise, there are uncertainties in the transfer orbit parameters.

Third, satellite trajectories are described by nonlinear differential equations. With orbital transfers, these differential equations change, and we obtain a system description as a nonlinear hybrid automaton. The software tools available for computing the reach set of such automata are limited, and thus, we resort to overapproximating the reach set. To address these challenges, we present three abstraction techniques.

Sequence of abstractions: Satellite orbits exhibit periodic motion, so the angular position of the satellite can be bounded between 0 and 2π. The transfer orbit parameters are determined by numerical methods and orbit determination measurements use noisy sensors. Thus, an exact transfer orbit may not be known, so we develop an abstraction for parameter uncertainty. The concrete model nondeterministically specifies the movement of the satellite along all (infinitely many) transfer orbits. That is, there may be infinitely many modes of the concrete hybrid automaton. Since the active satellite stays in the transfer orbit for a short period of time—an upper time bound is an input to Lambert's problem—we aggregate the motion along all such transfer orbits into a single mode of the hybrid system where the continuous evolution is defined by differential and algebraic equations. To accomplish this, we exploit monotonicity of the transfer orbit dynamics. For computing overapproximations of the reach set, nonlinear dynamics can be overapproximated by linear or rectangular hybrid automata. We employ the (now standard) *hybridization* technique [6, 10]. The state space of each mode of the original automaton is partitioned into a set of zones $\mathcal{Z}$, and within each zone $Z \in \mathcal{Z}$, the nonlinear differential equation $\dot{x} = f(x)$ is abstracted by simpler dynamics.

Contributions: The abstraction methods we develop—particularly transfer orbit aggregation—allow us to perform verification that compensates for numerical errors in the methods used to solve problems without analytic solutions that frequently arise in astrodynamics. We developed an automated abstraction tool to work on the class of periodic hybrid automata used to model systems like the satellite case studies in this paper. The abstraction tool is fully automatic, generating inputs to existing reachability tools for hybrid automata (HyTech [15], PHAVer [12], and SpaceEx [13]), and allows us to automatically verify time-bounded safety properties. Specifically for the case studies, we verified conjunction avoidance and rendezvous for several realistic examples, such as non-coaxial orbits, non-coplanar orbits, low-earth orbits, medium earth orbits, geosynchronous orbits, and geostationary orbits. The experimental results

demonstrate the utility of different approximation methods and their associated complexities. The abstractions we defined are useful by themselves and can be applied independently or together for other systems that require hybridization, are periodic, or are dependent on numerical solutions. Finally, we believe that the family of nonlinear hybrid models presented here can serve as realistic benchmarks for future verification research.

Related work: Most prior work on formal verification of satellite systems requires manual reasoning, but we mention a couple of semi-automatic methods. The algebraic framework based on Gröbner, described in [14] and extended in [1], can be used to determine the global minimum and maximum separation between two satellites. In contrast, our technique provides guarantees about all reachable states up to a bounded time horizon. Other recent work uses verified integration methods and interval analysis for proving collision avoidance of satellite systems [21]. None of these works handle orbital transfers.

There are a variety of hybrid systems reachability algorithms. We use the hybridization method from [6], which was extended to handle larger classes of nonlinear dynamics in [10]. Another hybridization method is developed in [2], which was applied to a truck rollover example with nonlinear dynamics in [3]. There is some theoretical work on periodic hybrid systems [11], and some case studies from circuits use reachability analysis for periodic hybrid systems [4]. Our work does not use an on-the-fly hybridization approach like some of the works just referenced, but we believe this was reasonable due to the periodicity of the examples studied.

2 Astrodynamics and Hybrid Systems Background

In this paper, a *satellite* is an object moving around the Earth under the influence of the latter's gravitational force. By Kepler's first law, the orbit of a satellite is an ellipse with the Earth at one of the foci, called the *main focus*, and thus the satellite remains in the same plane in 3-dimensional space.[4] Different orbits may or may not be coplanar or coaxial. Given the masses of the Earth and the satellite, and the relative position and velocity of the satellite (with respect to Earth), the orbit is uniquely defined.

Fixing an orbit, a satellite's motion in polar coordinates is given by the following equation, which captures Kepler's law of equal areas:

$$\dot{\nu} = f(\nu, p, e) = \sqrt{\frac{\mu}{p^3}}(1 + e\cos\nu)^2, \tag{1}$$

where ν is the angle of the satellite with respect to the major axis as measured from the main focus (known as the *true anomaly*), e is the *eccentricity*, $p = a(1 - e^2)$ is called the *semi-latus rectum*, a is the *semi-major axis*, and μ is the geocentric gravitational parameter. See Fig. 1 for a graphical depiction of these quantities.

[4] Generally, an orbit is some conic section, but we assume orbits are circular or elliptical (the eccentricity e of the orbit satisfies $0 \leq e < 1$).

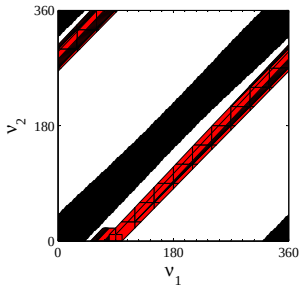

Fig. 3. Verification of conjunction avoidance over a single period for $d = 5000$km. The set $P_d(o_1, o_2)$ of (ν_1, ν_2) points where the distance between the two orbits is at most d is shown in black, and the time-bounded reach set is in red. The orbits are described by the parameters $e_1 = 0.05$, $p_1 = 7074$km, $e_2 = 0.10$, and $p_2 = 7748$km.

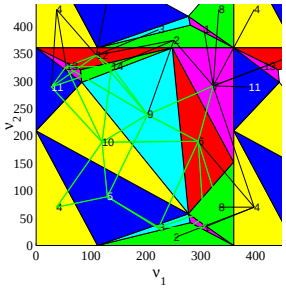

Fig. 4. Visualization of abstract system $\mathcal{B}$. The polygons due to the hybridization abstraction $\mathcal{A}_3$ partition the state space, and green lines are transitions between partitions. Black lines between centers of partitions are quotient transitions due to $\mathcal{A}_1$. Partitions on the post-state of a quotient transition are duplicated (e.g., see blue triangle labeled 11).

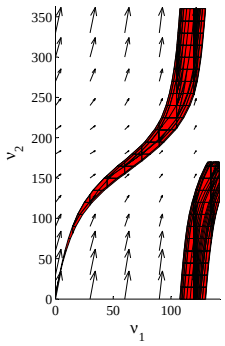

Fig. 5. $\mathsf{Reach}^{\delta}_{\mathcal{B}}$ for a pair of elliptical orbits with parameters $e_1 = 0.33$, $a_1 = 2$, $e_2 = 0.5$, and $a_2 = 1$. Black arrows are a vector field of the nonlinear dynamics. Error due to overapproximation of dynamics grows with time. Partition size was 15×15 degrees.

We refer the interested reader to [22, 8, 7] for derivations of this equation. Given an angle ν, Cartesian coordinates of the satellite are specified by

$$r = \frac{p}{1 + e\cos\nu},\ x = r\cos\nu,\ \text{and}\ y = r\sin\nu. \tag{2}$$

We consider verification of pairs of satellites performing the rendezvous operation (refer to Fig. 1). One passive and one active satellite each begin in respective initial orbits. In order to rendezvous with the passive satellite, when the active satellite arrives at a certain pre-calculated angular position, it switches (by firing its thrusters) to a transfer orbit. We will verify properties related to the proximity of the two satellites measured by their Euclidean distance in 3-dimensional space. Given two orbits o_1, o_2, and a distance threshold d, we define the set $P_d(o_1, o_2) \subseteq \mathbb{R}^2$ to be all (ν_1, ν_2) values at which the distance between the orbits is at most d. For coplanar orbits,

$$P_d(o_1, o_2) \triangleq \{(\nu_1, \nu_2)\ :\ ||(x_1, y_1) - (x_2, y_2)|| \leq d\} \tag{3}$$

where $||\cdot||$ is the 2-norm, and the Cartesian coordinates of each point on the orbit are determined by (2). See Fig. 3 for an example of this set. For non-coaxial and non-coplanar orbit pairs, the expression for $P_d(o_1, o_2)$ is analogous, albeit more complex.[5]

While thrusters typically actuate by burning over an interval of time, it is standard practice to model the actuation as an instantaneous change in dynamics due to the short duration of this burn time compared with the timescales involved in orbital motion. However, we note that approaches have been formulated to consider these finite-duration effects [20]. To rendezvous with the

[5] Descriptions of non-coaxial and non-coplanar orbits require the introduction of more orbital parameters, which for brevity we chose not to do, but we note that all the methods presented in this paper apply for non-coaxial and non-coplanar orbits.

passive satellite, usually the active satellite performs two burns. The first burn puts the active satellite on an intermediate *transfer orbit* that intersects the passive satellite's orbit. This burn is modeled as an instantaneous switch from the initial orbit parameters (e_I, p_I) to the transfer orbit parameters (e_T, p_T), and causes an instantaneous switch in the dynamics of $\dot{\nu}_2$ in (1). The second burn makes the active and passive satellites' orbits coincide and is modeled by another switch. One way to determine the transfer orbit parameters is by solving a problem called *Lambert's problem*, which is discussed in more detail in Section 4. Next, we discuss how such orbital transfers can naturally be modeled in the hybrid automata framework.

A *hybrid automaton (HA)* is a (possibly nondeterministic) state machine with state that can evolve both instantaneously (through *discrete transitions*) and over intervals of time (according to *trajectories*). In the satellite system model, the continuous variables of the HA model the angular positions of the satellites, and the discrete variables model the orbital parameters. The HA of Fig. 2 shows a two-burn rendezvous maneuver described earlier. Informally, when the HA is in a certain location (shown by the ellipses), the satellites move along specific orbits. That is, their angular positions evolve according to the differential equations corresponding to that location. The discrete transitions (shown by arrows) model the instantaneous burns.

The HA models the angular positions ν_1, ν_2 of two satellites. The passive satellite (ν_1) always moves along the same orbit specified by constant semi-latus rectum p_1 and eccentricity e_1. The active satellite (ν_2) begins in an initial orbit specified by parameters p_I and e_I. If the *guard* predicate G_{IT} is satisfied, then the active satellite must execute a burn that puts it on a transfer orbit. The transfer orbit is specified by the *reset map* R_{IT} that changes the valuations of p_2, e_2, and ν_2. Resetting the variable ν_2 is needed to model transfer orbits that are not coaxial with the initial orbit. That is, the same point in Cartesian coordinates may no longer correspond to the same polar coordinates because the transfer orbit may not be coaxial with the initial orbit. The second burn is modeled in an identical fashion, and sequences of burns can be modeled similarly.

Now we define the HA formally based on previous HA modeling frameworks [5, 18, 16]. Variables are associated with types and are used as names for state components, such as the angular positions and the orbital parameters. For a set of variables V, a valuation $\mathbf{v}$ is a function that maps each variable $v \in V$ to a point in its type. The set of all possible valuations is $val(V)$. For a valuation $\mathbf{x}$, we use $\mathbf{x}.x$ to denote the value of the variable $x \in V$.

The *concrete HA* is a tuple $\mathcal{A} \triangleq \langle V, Q, \Theta, Edg, Grd, Rst, Flow, Inv \rangle$, where: (a) $V \triangleq \{X, loc, p_1, e_1, p_2, e_2\}$. V is a set of variables, where $X \triangleq \{\nu_1, \nu_2\}$ are real-valued continuous variables, p_1, e_1, p_2, and e_2 are real-valued discrete variables modeling the orbit parameters, and $loc \in L$ is a discrete variable of type $L \triangleq \{I, T, R\}$, where elements represent respectively the initial, transfer, and rendezvous orbits. (b) $Q \triangleq val(V)$ is the set of states. For a state $\mathbf{x} \in Q$, the valuation of $\mathbf{x}.loc$ is called the *location*; along with the valuations of the discrete variables p_1, e_1, p_2, e_2, it describes the discrete state. The valuation of the con-

tinuous variables X, that is $\{\mathbf{x}.x : x \in X\}$, is called the *continuous state* and is referred to as $\mathbf{x}.X$. (c) $\Theta \subseteq Q$ is a set of *initial states*. (d) $Edg = \{(I,T),(T,R)\}$ is the set of *edges*. (e) $Grd : Edg \to Q$ is a function that associates a *guard* (a valuation of V that must be satisfied) with each edge. The guards are shown in Fig. 2. $Grd((I,T)) \triangleq G_{IT}(\nu_1,\nu_2)$ and $Grd((T,R)) \triangleq G_{TR}(\nu_1,\nu_2)$; that is, they are left as parameters. (f) $Rst : Edg \to (Q \to 2^Q)$ is a function, called the *reset map*, associated with each edge. A reset map associates a set of states with each edge: $Rst((I,T)) \triangleq \nu_2' = R_{IT}(\nu_1,\nu_2)$ and $Rst((T,R)) \triangleq \nu_2' = R_{TR}(\nu_1,\nu_2)$. (g) $Flow : L \to (Q \to 2^Q)$ associates a *flow map* with each location. Here, for $l \in L$ and where f is from (1), we have $Flow(l) = [f(\nu_1,p_1,e_1); f(\nu_2,p_2,e_2)]$. (h) $Inv : L \to 2^Q$ associates an *invariant* with each location. Here we assume urgency, so $Inv(I) = \mathbb{R}^2 \setminus Grd((I,T))^\circ$ and $Inv(T) = \mathbb{R}^2 \setminus Grd((T,R))^\circ$, where, for a real-valued set R, R° is the interior of R.

The semantics of HA $\mathcal{A}$ are defined in terms of sets of *transitions* and *trajectories*. The set of transitions $\mathcal{D} \subseteq Q \times Q$ is defined as follows. We have $(\mathbf{v},\mathbf{v}') \in \mathcal{D}$ if and only if for $e = (\mathbf{v}.loc, \mathbf{v}'.loc)$, (a) $e \in Edg$, (b) $\mathbf{v} \in Grd(e)$, and (c) $\mathbf{v}' \in Rst(e)(\mathbf{v}.X)$. A *trajectory* for $\mathcal{A}$ is a function $\tau : [0,t] \to Q$ that maps an interval of time to states such that the following hold. (a) For all $t' \in [0,t]$, $\tau(t').loc = \tau(0).loc$, that is, the discrete state remains constant. (b) $(\tau \downarrow X)$, that is, the restriction of τ to X is a solution of the differential equation specified by the flow function $\dot{X} = Flow(\tau(0).loc)(\tau(0))$. (c) For all $t' \in [0,t]$, $\tau(t') \in Inv(\tau(0).loc)$. The set of all the trajectories of $\mathcal{A}$ is written $\mathcal{T}$.

An *execution* of $\mathcal{A}$ is a sequence $\alpha = \tau_0\tau_1\ldots$, such that (a) each $\tau_i \in \mathcal{T}$, (b) for each i, $(\tau_i(t), \tau_{i+1}(0)) \in \mathcal{D}$, where t is the right endpoint of the domain of τ_i, and (c) $\tau_0 \in \Theta_0$. The set of all executions of $\mathcal{A}$ is denoted by $\mathsf{Execs}_\mathcal{A}$. A state $\mathbf{v} \in Q$ is said to be *reachable* if there exists a closed execution α that ends at $\mathbf{v}$. The set of all reachable states of $\mathcal{A}$ is denoted by $\mathsf{Reach}_\mathcal{A}$. The set of states reachable of $\mathcal{A}$ within δ time is denoted by $\mathsf{Reach}^\delta_\mathcal{A}$ and is called the set of bounded-time reachable states (see Fig. 5 as an example). We define $\mathsf{Reach}_\mathcal{A}(t)$ as the set of states that are reachable by executions of $\mathcal{A}$ at exactly t time, and for $t \leq \delta$, $\mathsf{Reach}^\delta_\mathcal{A}(t)$ is defined analogously.

We write $\mathcal{D}_\mathcal{A}$, $\mathcal{T}_\mathcal{A}$, $Rst_\mathcal{A}$, $V_\mathcal{A}$, etc., for the components of $\mathcal{A}$ if the automaton is not clear from context. Similarly, when necessary to disambiguate components of HA $\mathcal{A}$ from those of HA $\mathcal{B}$, we use subscripts such as $Q_\mathcal{A}$, $Inv_\mathcal{A}$, $Rst_\mathcal{B}$, etc. Given a pair of HA $\mathcal{A}$ and $\mathcal{B}$, $\mathcal{B}$ is said to be an *abstraction* for $\mathcal{A}$ if $\mathsf{Execs}_\mathcal{A} \subseteq \mathsf{Execs}_\mathcal{B}$. It follows that if $\mathcal{B}$ is an abstraction of $\mathcal{A}$, then $\mathsf{Reach}_\mathcal{A} \subseteq \mathsf{Reach}_\mathcal{B}$. Also, if $\mathcal{B}$ is safe with respect to some property (set), then so is $\mathcal{A}$.

3 Abstractions and Analysis

To verify conjunction avoidance and rendezvous properties, we compute bounded reach sets, which is difficult for nonlinear HA. In this section, we describe three independent abstractions of periodic, nonlinear HA (quotienting, transfer orbit aggregation, and hybridization), and then apply their composition.

Quotienting: The quantities ν_1 and ν_2 model the angular position of the satellites on their orbits, which are periodic with period 2π. We define a quotient HA $\mathcal{A}_1$ based on an equivalence relation $\sim$:

$$\mathbf{x} \sim \mathbf{x}' \iff \exists k_1, k_2, \forall\, i \in \{1,2\},\ \mathbf{x}.\nu_i = \mathbf{x}'.\nu_i + k_i 2\pi.$$

Using $\sim$, we reduce the unbounded state space to a bounded one by adding transitions to each mode of the concrete HA $\mathcal{A}$. If some ν_i reaches the 2π boundary, it is reset to 0. These are the only edges and resets we add, since $\dot{\nu}_1 > 0$ and $\dot{\nu}_2 > 0$ (the angular positions are monotonically increasing), but in general, it may be necessary to add transitions when $\nu_i = 0$ if $\dot{\nu}_i < 0$. $\mathcal{A}_1$ is bisimilar to $\mathcal{A}$.

Transfer orbit aggregation: Solving the Lambert problem yields a unique transfer orbit, where the trajectory of the active satellite would begin from a ν_2 angle called the *burn point*. There is also a constraint on the passive satellite's angle so that the two satellites can rendezvous, so the burn point is a pair of (ν_1, ν_2) values. However, the burn point is not known precisely, so a burn actually takes place within a range of (ν_1, ν_2) values. Each (ν_1, ν_2) pair will place the active satellite on a slightly different transfer orbit. Thus, the transfer mode must take into account a set of different possible transfer orbits. The following abstraction aggregates this set of transfer orbit parameters into a single location of the HA.

First, we define the set of possible transfer orbits that could be reached by burning at different points.

Definition 1. *For any set $\mathcal{O}$ of transfer orbit parameter pairs $o \triangleq (p, e) \in \mathcal{O}$, consider $R_i \subseteq \mathbb{R}$ for $i \in \{1, 2, \ldots, k\}$ such that $\cup_i R_i = \mathbb{R}$, where $\forall i \in \{1, 2, \ldots, k\}$,*

- *(i) $\exists (p_{min}, e_{min})$ such that $\forall o = (p, e) \in \mathcal{O}$, $\forall \nu_2 \in R_i$, we have $f_{min}(\nu_2) \triangleq f(\nu_2, p_{min}, e_{min}) \leq f(\nu_2, p, e)$, and*
- *(ii) $\exists (p_{max}, e_{max})$ such that $\forall o = (p, e) \in \mathcal{O}$, $\forall \nu_2 \in R_i$, we have $f_{max}(\nu_2) \triangleq f(\nu_2, p_{max}, e_{max}) \geq f(\nu_2, p, e)$.*

That is, $f_{min}(\nu_2)$ and $f_{max}(\nu_2)$ are lower and upper bounds of the $\dot{\nu}_2$ dynamics for a particular region R_i.

Given a collection $\{R_i\}$ that satisfies the requirements in Definition 1, the *HA with transfer orbit aggregation* is a tuple $\mathcal{A}_2 \triangleq \langle V, Q, \Theta, Edg, Grd, Rst, Flow, Inv \rangle$, where: (a) $V = V_{\mathcal{A}}$, (b) $Q = Q_{\mathcal{A}}$, (c) $\Theta = \Theta_{\mathcal{A}}$, (d) $Edg = Edg_{\mathcal{A}}$, (e) $Grd : Grd_{\mathcal{A}}$, and (f) $Rst : Rst_{\mathcal{A}}$, but now the guard and reset maps between modes correspond to sets of ν_1, ν_2 values. (g) $Flow_{\mathcal{A}_2}$: Using the set of all (p, e) pairs of $\mathcal{O}$, the $\dot{\nu}_2$ dynamics for the active satellite in the transfer mode are defined piecewise over all R_i such that for each R_i, we have $\dot{\nu}_2 \in [f_{min}(\nu_2), f_{max}(\nu_2)]$.

The dynamics of $\mathcal{A}_2$ and $\mathcal{A}$ are identical except when the active satellite is in the transfer mode. For that mode, the dynamics corresponding to any execution of $\mathcal{A}$ are contained within the dynamics of $\mathcal{A}_2$ by construction, since $\mathcal{A}_2$ creates piecewise upper and lower bounds on $\dot{\nu}_2$. Thus we have that $\mathcal{A}_2$ is an abstraction of $\mathcal{A}$.

Hybridization: Our approach for both verification problems relies on computing the reachable states $\mathsf{Reach}_{\mathcal{A}}$ of the HA $\mathcal{A}$. Since the software tools for

computing the reach set of nonlinear HA are not as well-developed as those for linear and rectangular HA, we abstract the given nonlinear HA by a HA with simpler dynamics. We employ the *hybridization* approach [6, 10], where the state-space of $\mathcal{A}$ is partitioned into a finite number of zones (see the polygons in Fig. 4). The nonlinear dynamics are conservatively approximated within each zone with simpler dynamics—in our case either (a) rectangular or (b) linear (affine) dynamics.

Given HA $\mathcal{A}$ and a partition function P that returns, for each location $l \in L$, a partition $\{I_1, \ldots, I_k\}$ such that $\cup_{j=1}^{k} I_j = Inv(l)$, we define the *hybridization abstraction* as the tuple $\mathcal{A}_3 \triangleq \langle V, Q, \Theta, Edg, Grd, Rst, Flow, Inv \rangle$, where: (a) $V = V_{\mathcal{A}} \cup zone$, where $zone$ is a discrete variable of type $Z_l = \{1, \ldots, k\}$ and identifies the partitions of each mode. (b) $Q = val(V_{\mathcal{A}_3})$ is the set of states. Now, for $\mathbf{x} \in Q$, the valuations of $\mathbf{x}.loc$, $\mathbf{x}.zone$, and the orbit parameter variables describe the discrete state. (c) $\Theta \subset Q$. (d) $Edg \subseteq (L \times Z) \times (L \times Z)$ is defined as follows: $((l, z), (l', z')) \in Edg$ if and only if either (i) $l' = l$ and $I_{z'}$ is adjacent to I_z, or (ii) $l' \neq l$ and I_z is contained in $I_{z'}$. (e) $Grd : Edg \to Q$ is defined as $Grd(((l, z), (l', z'))) = Inv_{\mathcal{A}}(l) \cap I_z$. (f) $Rst : Edg \to (Q \to 2^Q)$ is defined as (i) if $l = l'$, then the reset is the identity, and (ii) $Rst_{\mathcal{A}}(l, l')$, otherwise. (g) $Flow : (L \times Z) \to (Q \to 2^Q)$ is the flow map defined as follows. For each satellite $i \in \{1, 2\}$, location l, and zone z, we associate either (i) rectangular differential inclusions: $\dot{\nu}_i \in [a_i, b_i]$ for

$$a_i = \min_{\mathbf{x}.\nu_i \in I_z} Flow_{\mathcal{A}}(l)(\mathbf{x}) \text{ and } b_i = \max_{\mathbf{x}.\nu_i \in I_z} Flow_{\mathcal{A}}(l)(\mathbf{x}),$$

or (ii) affine (linear) differential inclusions: $\dot{\nu} = A\nu + b \pm \epsilon$, for

$$A = \nabla Flow_{\mathcal{A}}(l)(\mathbf{x})|_c \cdot (\nu - c),\ b = f(c),\ \epsilon = \max_{\mathbf{x}.\nu \in I_z} ||Flow_{\mathcal{A}}(l)(\mathbf{x}) - A\nu - b||,$$

where $c \in \mathbb{R}^2$ is the centroid of z, and $\nabla Flow_{\mathcal{A}}(l)(\mathbf{x})|_c$ is the Jacobian evaluated at c of $Flow_{\mathcal{A}}(l)(\mathbf{x})$. (h) $Inv : (L \times Z) \to 2^{val(X)}$ is $Inv_{\mathcal{A}_3}(l, z) \triangleq Inv(l) \cap I_z$.

By construction, the dynamics of $\mathcal{A}$ are contained in the conservative over-approximation, and a proof that $\mathcal{A}_3$ is an abstraction of $\mathcal{A}$ appears in [6]. Each of the individual abstractions are sound and can be implemented independently of one another. Thus, applying the abstractions $\mathcal{A}_1$, $\mathcal{A}_2$, and $\mathcal{A}_3$ sequentially to $\mathcal{A}$ yields another HA called $\mathcal{B}$ (visualized in Fig. 4), which is an abstraction of $\mathcal{A}$, since the composition of abstractions is sound.

Impossibility of unbounded model checking: Consider two arbitrary orbits o_1 and o_2 with periods T_1 and T_2. These two orbits are said to be *relatively periodic* if $\frac{T_1}{T_2}$ is rational; otherwise, they are said to be *incommensurate*. For circular orbits, the right-hand side of (1) reduces to a constant, and consequently, the reach set can be computed exactly. However, if the ratio of the orbits' periods is irrational, this is impossible. The proof of this follows from the mathematical result that the reach set of a point with irrational slope on the unit torus (or the unit square with billiards reflections at edges) is dense [19].

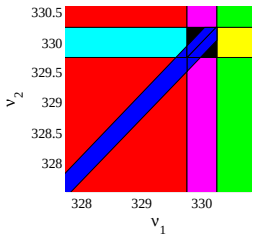

Fig. 6. Reach set of the initial orbit is plotted in blue. Black region is the expanded Grd_{IT} determined by transfer orbit aggregation around the original burn guard $(330, 330)$.

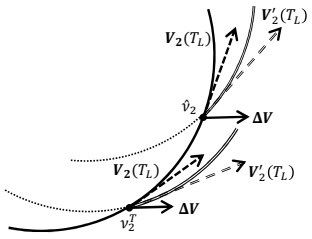

Fig. 7. Transfer orbit calculation: Lambert burn vector $\boldsymbol{\Delta V}$ applied to the original burn angle ν_2^T and neighboring point $\hat{\nu}_2$. Resulting velocity vectors and transfer orbits shown.

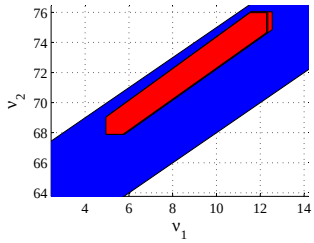

Fig. 8. Example verification of rendezvous for $d = 500$km. Blue set is the distance set Γ_d. Red set is the time intersection reach set, $\mathsf{Reach}_{\mathcal{B}}^{\Delta}(t)$ for $t = T_R$, the rendezvous time.

4 Computation of Abstractions

In this section, we describe how the transfer orbit aggregation abstraction is computed in our abstraction tool. We use boldface to indicate vectors.

First, we summarize how an ideal thrust vector $\boldsymbol{\Delta V}$ is computed by numerically solving Lambert's problem. Then, we show how $\boldsymbol{\Delta V}$ is applied to points nearby the original burn point. This yields uncountably infinitely many transfer orbits, each denoted by o_i, where $i \in \mathcal{O}$ for an uncountably infinite index set $\mathcal{O}$. We collapse this set of transfer orbits to a single mode by overapproximating the dynamics to include all possible transfer orbits.

Computation of ideal thrust vector $\boldsymbol{\Delta V}$: To calculate the orbit of the active satellite following a burn, we use an equivalent representation of the orbit dynamics—the position and velocity of the satellite in 3-dimensional Cartesian space. Recall that a satellite's orbit is completely described by (1) with parameters p, e, and angular position ν. In Cartesian coordinates, the satellite is described by a position vector $\boldsymbol{r}$ and velocity vector $\boldsymbol{V}$. We will use both of these representations in the following procedure to calculate the transfer orbit.

Let T_L be the time when the (instantaneous) burn occurs, and let the angular positions of the two satellites at T_L be $(\nu_1(T_L), \nu_2(T_L))$. Let the time-to-rendezvous be T_R. The next sequence of steps describes how to compute the magnitude and direction of force that the burn applies to the active satellite. (a) $\boldsymbol{r}_i(T_L)$ and $\boldsymbol{V}_i(T_L)$ are computed at the passive and active satellites' initial positions $\nu_i(T_L)$. (b) Given the time of transfer T_R, $\nu_1(T_L + T_R)$ is computed by numerical integration of (1), and then the rendezvous location, $\boldsymbol{r}_1(T_L + T_R)$, is computed using $\nu_1(T_L + T_R)$. (c) The active satellite's states $\boldsymbol{r}_2(T_L)$ and $\boldsymbol{V}_2(T_L)$, and desired position for rendezvous $\boldsymbol{r}_2(T_L + T_R) = \boldsymbol{r}_1(T_L + T_R)$, are used to solve Lambert's problem to determine the velocity $\boldsymbol{V}_2'(T_L)$ that defines the transfer orbit. We then convert this velocity $\boldsymbol{V}_2'(T_L)$ to the transfer orbit parameters e_T and p_T needed to achieve rendezvous.

From the transfer orbit parameters, the required change in velocity at T_L is $\boldsymbol{\Delta V} = \boldsymbol{V}_2'(T_L) - \boldsymbol{V}_2(T_L)$. In reality, the time of and angular positions at burning are not known exactly, and as a result, the calculated $\boldsymbol{\Delta V}$ puts the active satellite on one of a collection of transfer orbits.

Expanding the Lambert burn angle to a range of angles: To construct $\mathcal{A}_2$ for rendezvous, we have to instantiate Grd_{IT}. Consider a point representing the minimum energy burn in the ν_1, ν_2 state space. Uncertainties in initial conditions, measurements, and numerical errors in position estimation cause the concrete system to have a larger guard. Thus, this is also incorporated into the abstract system. As a result, a given execution of the automaton may perform the burn within a set of different angular positions (and velocities). Also, the partitioning scheme around this minimum burn point must be adjusted to accommodate the larger guard, as shown in Fig. 6. Next, we outline the details of calculating transfer orbits of $\mathcal{O}$ for points within a small neighborhood of an ideal Lambert burn point. Let $(\nu_1^T, \nu_2^T) = G_{IT}$ be the ideal Lambert burn point. In general, we will add $\boldsymbol{\Delta V}$ to neighboring points to obtain a new $\boldsymbol{V}$ and then convert to the equivalent angular representation as shown in Fig. 7. The following calculations pertain only to the active satellite for some $\hat{\nu}_2$ location. Hence, we denote all initial orbit quantities with subscript I and transfer orbit quantities with subscript T, so $\boldsymbol{V}_2(T_L) = \boldsymbol{V}_I$ and $\boldsymbol{V}_2'(T_L) = \boldsymbol{V}_T$.

(a) Determine nearby points within guard set of the active satellite: $\hat{\nu}_2 \in \Lambda \triangleq [\nu_2^T - \epsilon, \nu_2^T + \epsilon]$. (b) For $\hat{\nu}_2$, calculate the position and velocity vectors in Cartesian coordinates using the orbital parameters of the initial orbit. For converting between the angular representation and Cartesian representation, we introduce an eccentricity vector $\boldsymbol{e_I}$ and angular momentum vector $\boldsymbol{h_I}$ [22]. The vectors $\boldsymbol{e_I}$ and $\boldsymbol{h_I}$ give direction with respect to the axes of the elliptical orbit. We write e_I, h_I, etc., without vector boldface, as the magnitude of the corresponding vector. The conversion is done by computing: $p_I = a_I(1 - e_I^2)$, $\boldsymbol{e_I} = [e_I;\ 0;\ 0]$, $\boldsymbol{h_I} = [0;\ 0;\ \sqrt{\mu p_I}]$, and

$$\boldsymbol{V}_I = \frac{\mu}{h_I^2}\boldsymbol{h_I} \times (\boldsymbol{e_I} + [\cos(\hat{\nu}_2);\ \sin(\hat{\nu}_2);\ 0]),$$
$$r_I = \frac{p_I}{1 + e_I \cos(\hat{\nu}_2)}, \text{ and } \boldsymbol{r}_I = r_I[\cos(\hat{\nu}_2);\ \sin(\hat{\nu}_2);\ 0].$$

(c) Now, add the Lambert burn vector $\boldsymbol{\Delta V}$ corresponding to the angle ν_2^T to the velocity vector $\boldsymbol{V}_I$ at $\hat{\nu}_2$, $\boldsymbol{V}_T = \boldsymbol{V}_I + \boldsymbol{\Delta V}$. (d) From the position (note that $\boldsymbol{r}_I = \boldsymbol{r}_T$) and resultant velocity vectors at $\hat{\nu}_2$, calculate the corresponding transfer orbit parameters:

$$\boldsymbol{h}_T = \boldsymbol{V}_T \times \boldsymbol{r}_T, \qquad \boldsymbol{e}_T = \frac{1}{\mu}(\boldsymbol{V}_T \times \boldsymbol{h}_T) - \frac{\boldsymbol{r}_T}{r_T}, \qquad a_T = \frac{h_T}{\mu}(1 - e_T^2),$$
$$p_T = a_T(1 - e_T^2), \text{ and} \qquad \nu_2' = \arctan\left(\frac{\boldsymbol{e}_T[2]}{\boldsymbol{e}_T[1]}\right),$$

where for a vector $\boldsymbol{x}$, the notation $\boldsymbol{x}[j]$ accesses the j^{th} component of that vector. Here, ν_2' is the reset value for ν_2, which corresponds to the angular shift in the coordinate frame of a single transfer orbit. Since there is a transfer orbit for each $\hat{\nu}_2 \in \Lambda$, the reset for ν_2 will be in a range defining the reset R_{IT}.

Now that we can calculate transfer orbits for points from Λ, there are two issues to address. First, the dynamics of the transfer mode in the abstraction $\mathcal{A}_2$ must include all possible transfer orbit dynamics. To address this, we revisit (1). The parameters e_T and p_T for the transfer orbit are now defined in terms of

$\hat{\nu}_2$ for $\hat{\nu}_2 \in \mathit{\Lambda}$. That is, $p(\hat{\nu}_2)$ and $e(\hat{\nu}_2)$ are functions representing all possible transfer orbit parameters. Thus, the nonlinear differential inclusion describing all transfer orbits of the active satellite is $\dot{\nu_2} = \sqrt{\mu/p(\hat{\nu}_2)^3}(1 + e(\hat{\nu}_2)\cos(\nu_2))^2$. In general, the definition of $\mathcal{A}_2$ requires the dynamics to be described by a function with upper and lower bounds. Thus, rectangular dynamics satisfy this definition, although we could use any appropriate upper and lower bounded function, e.g., the linear overapproximation used in hybridization. We construct rectangular dynamics for $\mathcal{A}_2$ by solving the following optimization problem:

$$\dot{\nu}_{2min} = \min_{\nu_2 \in R_i \wedge \hat{\nu}_2 \in \mathit{\Lambda}} f(\nu_2, p(\hat{\nu}_2), e(\hat{\nu}_2)),\ \dot{\nu}_{2max} = \max_{\nu_2 \in R_i \wedge \hat{\nu}_2 \in \mathit{\Lambda}} f(\nu_2, p(\hat{\nu}_2), e(\hat{\nu}_2)).$$

For a particular partition in the transfer mode, we first minimize or maximize $\cos(\hat{\nu}_2)$. Now, replacing $\cos(\nu_2)$ with this optimized value in $\dot{\nu_2}$ will allow optimization over the single variable $\hat{\nu}_2$.

The second issue is that since there are a continuum of possible transfer orbits, we must generalize the distance threshold set P_d from (3) that was previously defined for a single pair of orbits. If the active satellite is on one of many possible transfer orbits, then to ensure the rendezvous property is satisfied, the satellites must be within d for each of these possible orbits. We ensure this by calculating a distance set $\mathit{\Gamma}_d$ that holds for every transfer orbit. The active satellite's position is defined in terms of the functions $p(\hat{\nu}_2), e(\hat{\nu}_2)$ such that:

$$\mathit{\Gamma}_d \triangleq \{(\nu_1, \nu_2) \ : \ ||(x_1, y_1) - (x_2(p(\hat{\nu}_2), e(\hat{\nu}_2)), y_2(p(\hat{\nu}_2), e(\hat{\nu}_2)))|| \leq d\}\,. \tag{4}$$

In practice, we form this as an optimization problem by maximizing the norm from (4) over $\hat{\nu}_2$ for any particular point (ν_1, ν_2) in the state space. If this maximum distance is within the threshold d, then for any transfer orbit, the active satellite at that point is within d of the passive satellite. For both of these issues, when there is not an analytic solution to the optimization, we can introduce an error bound ϵ to the function being optimized to preserve soundness. For instance, if ϵ is the maximum error in the optimization of the distance equation, we would compute $\mathit{\Gamma}_{d-\epsilon}$ to ensure that any potential verified rendezvous satisfies the actual distance threshold d.

We now summarize the procedure for verifying rendezvous maneuvers. With a set of initial conditions for ν_1, ν_2, initial orbits o_1, o_2, and a Lambert burn point, the abstract HA $\mathcal{B}$ is computed as just described. Next, using $\mathcal{B}$ as input to HyTech, PHAVer, or SpaceEx, calculate $\mathsf{Reach}^{\delta}_{\mathcal{B}}$ for a bounded time δ. Then, take a time intersection $\mathsf{Reach}^{\delta}_{\mathcal{B}}(t)$ for a possible rendezvous time $t < \delta$. If $\mathsf{Reach}^{\delta}_{\mathcal{B}}(t) \subseteq \mathit{\Gamma}_d$, then the reachable set of states at time t is within the distance threshold d. An example $\mathsf{Reach}^{\delta}_{\mathcal{B}}(t)$ and $\mathit{\Gamma}_d$ are shown in Fig. 8.

5 Experimental Results

We present experimental results for verifying conjunction avoidance and rendezvous using the three abstractions applied to the original system.

Once the abstract system $\mathcal{B}$ is constructed using our tool, the conjunction avoidance and rendezvous properties can be verified by computing $\mathsf{Reach}^{\delta}_{\mathcal{B}}$ for

Table 1. Rendezvous experiments for $d = 500$km, $\epsilon = 0.25$ for guard Λ, and partition size 20×20 degrees. AT is abstraction run time (sec). PT is PHAVer run time (sec). RT is the time interval of rendezvous (sec), with the burn occurring in time at the lower bound of this interval. The underline and overlined parameters e_T or p_T are respectively the min and max of the nondeterministic parameter values.

Initial (ν_1, ν_2)	Guard (ν_1^T, ν_2^T)	Initial Orbit $(e_1, p_1[km], e_I, p_I[km])$	Transfer Orbit $(\underline{e_T}, \overline{e}_T, \underline{p_T}[km], \overline{p}_T[km])$	AT (s)	PT (s)	RT (s)
$(270, 267.5)$	$(330, 330)$	$(0, 6718, 0.05, 7340)$	$(0.05849, 0.05853, 6766, 6769)$	811	3.01	$(950, 1200)$
$(250, 246.5)$	$(330, 330)$	$(0, 6718, 0.05, 7340)$	$(0.05849, 0.05853, 6766, 6769)$	811	3.23	$(1050, 1300)$
$(300, 299)$	$(333, 333)$	$(0.05, 7074, 0.10, 7748)$	$(0.06468, 0.06486, 7114, 7116)$	801	3.4	$(500, 1250)$
$(300, 299)$	$(327, 327)$	$(0, 6718, 0.10, 7748)$	$(0.06186, 0.06202, 6982, 6984)$	834	3.37	$(440, 990)$

some bounded time δ. Our prototype tool is written in Matlab and experiments were carried out on a modern laptop running Windows 7 with 4GB RAM and a 2.0GHz dual-core Intel i5 processor. We used PHAVer [12] and SpaceEx [13] for verification of $\mathcal{B}$.[6] SpaceEx runs in a virtual machine, and we also ran HyTech and PHAVer in an Ubuntu virtual machine. Overall, our results using HyTech, PHAVer, and SpaceEx suggest that tools allowing for relatively complex discrete dynamics and large numbers of locations need to be complemented with more scalable continuous reachability methods. We previously showed conjunction avoidance verification for one set of parameters in Fig. 3. Our test cases included Low Earth Orbits (LEO, altitude below 2000km), Medium Earth Orbits (MEO, 2000km to 35,786km), Geo Stationary (GEO), and Geo Synchronous (GSO) orbits with varying eccentricities. We were able to verify rendezvous for LEOs with eccentricities between 0 and 0.1.

Table 1 shows some successful rendezvous test cases, which used a rendezvous distance $d = 500$km, rectangular overapproximation of dynamics, and PHAVer. The first column is the initial state of the continuous variables. The second column is the ideal guard G_{IT} around which the abstracted guard Λ is built. The initial orbit parameters are shown as well as the parameter ranges that define the continuum of transfer orbits in the second mode. The RT column shows the time intersection of the reach set where the rendezvous was satisfied. To verify smaller rendezvous distances, smaller partitioning sizes can be used. This will minimize the error accumulated in the approximation, but will result in increased abstraction and reach set computation time. The bounded reach set, $\mathsf{Reach}^{\delta}_{\mathcal{B}}$, is not completely contained in Γ_d, and only its intersection for a range of times ($\mathsf{Reach}^{\delta}_{\mathcal{B}}(t)$ for $t \in [T_R - \rho, T_R + \rho]$) is completely contained. For instance, one input to Lambert's problem is the time T_R for rendezvous to occur, and we can verify rendezvous for a range of times ρ around T_R.

Table 2 compares different hybridization schemes—rectangular versus linear overapproximations of dynamics—for conjunction avoidance. We fix the partitioning of the hybridization and are comparing only rectangular versus linear dynamics for the same partition shape and size. Usually, the reach sets from linear overapproximation are smaller than rectangular. However, the support function algorithm implemented in SpaceEx allows the user to configure the amount of error in the overapproximation. Lower error comes at the cost

[6] We found HyTech [15] to be unusable for elliptical orbits due to numerical overflows.

Table 2. Reachability experiments for different overapproximation techniques. Initial condition is $(\nu_1, \nu_2) = (0, 0)$. RA and AA columns are the abstraction times in seconds for rectangular and affine dynamics, respectively. PR, SR, and SA columns are, respectively, the run time in seconds of PHAVer with rectangular dynamics, SpaceEx with rectangular dynamics, and SpaceEx with affine (linear) dynamics. The number subscript for the SpaceEx runs determine the sampling time used in the reachability algorithm. These experiments ran until the time bound T equal to the satellite period.

Parameters $(e_1, p_1[km], e_2, p_2[km])$	Partition Size	RA	AA	PR	SR_{20}	SR_{100}	SA_{20}	SA_{100}
$[0.05, 7056, 0.10, 7670]$	60 x 60	4.42	5.89	0.26	979	249	193	140
$[0.10, 7467, 0.10, 7670]$	60 x 60	9.8	9.92	0.35	1076	263	384	191

of higher runtime, and we summarize runtime comparisons in Table 2. We can decrease this runtime cost by configuring SpaceEx, but this may come at the expense of the rectangular overapproximation being as good, if not better, than the linear overapproximation.

6 Conclusion and Future Work

In this paper, we developed abstraction techniques used to enable automatic verification of bounded-time safety properties for nonlinear satellite systems. The abstractions account for uncertainties in observation times, sensor measurements, and thrusting. We also showed that the unbounded model-checking of incommensurate orbits is impossible. However, the reach set for circular commensurate orbits can be computed exactly. While we do not have space to present it here, we (a) can verify unbounded properties of eccentric commensurate orbits by using forward and backward reachability techniques, and (b) have verified time-bounded safety properties for nearby satellites using the Clohessy-Wiltshire-Hill (CWH) dynamics in the ellipsoidal toolbox [17].

One of the primary roadblocks for analyzing more eccentric elliptical orbits or multiple-transfer satellite maneuvers is the granularity with which we are able to partition the state space. If we are able to approximate the dynamics over smaller intervals, we will be better equipped to analyze these more complex systems. An important feature yet to be taken advantage of is that the dynamics between the satellites is loosely coupled. A new approach we are exploring is to decompose the multi-satellite system into individual satellite automata, which would allow for much finer partitioning. With each automata containing a synchronized clock variable, we are developing algorithmic techniques that act on the individual reach sets to enable compositional verification of the global safety properties.

References

1. Allgeier, S.E., Fitz-Coy, N.G., Erwin, R.S., Lovell, T.A.: Metrics for mission planning of formation flight. In: Proc. AAS/AIAA Astrodynamics Specialist Conference. Girdwood, AK (Jul 2011)
2. Althoff, M., Stursberg, O., Buss, M.: Reachability analysis of nonlinear systems with uncertain parameters using conservative linearization. In: 47th IEEE Conference on Decision and Control (CDC). pp. 4042–4048 (Dec 2008)

3. Althoff, M., Le Guernic, C., Krogh, B.H.: Reachable set computation for uncertain time-varying linear systems. In: Hybrid Systems: Computation and Control (HSCC). pp. 93–102. ACM, New York, NY (2011)
4. Althoff, M., Rajhans, A., Krogh, B., Yaldiz, S., Li, X., Pileggi, L.: Formal verification of phase-locked loops using reachability analysis and continuization. In: IEEE/ACM International Conference on Computer-Aided Design (ICCAD) (2011)
5. Alur, R., Courcoubetis, C., Halbwachs, N., Henzinger, T.A., Ho, P.H., Nicollin, X., Olivero, A., Sifakis, J., Yovine, S.: The algorithmic analysis of hybrid systems. Theoretical Computer Science 138(1), 3–34 (1995)
6. Asarin, E., Dang, T., Girard, A.: Hybridization methods for the analysis of nonlinear systems. Acta Informatica 43, 451–476 (2007)
7. Bate, R.R., Mueller, D.D., White, J.E.: Fundamentals of Astrodynamics. Dover (1971)
8. Battin, R.: An introduction to the mathematics and methods of astrodynamics. AIAA education series, American Institute of Aeronautics and Astronautics (1999)
9. Bosse, A.B., Barnds, W.J., Brown, M.A., Creamer, N.G., Feerst, A., Henshaw, C.G., Hope, A.S., Kelm, B.E., Klein, P.A., Pipitone, F., Plourde, B.E., Whalen, B.P.: Sumo: Spacecraft for the Universal Modification of Orbits. In: Proc. of SPIE. vol. 5419, pp. 36–46 (2004)
10. Dang, T., Maler, O., Testylier, R.: Accurate hybridization of nonlinear systems. In: Hybrid Systems: Computation and Control (HSCC). pp. 11–20. ACM, New York, NY (2010)
11. Flieller, D., Riedinger, P., Louis, J.: Computation and stability of limit cycles in hybrid systems. Nonlinear Analysis: Theory, Methods, & Applications 64(2), 352–367 (2006)
12. Frehse, G.: PHAVer: Algorithmic verification of hybrid systems past HyTech. In: Hybrid Systems: Computation and Control (HSCC). pp. 258–273 (2005)
13. Frehse, G., Le Guernic, C., Donzé, A., Cotton, S., Ray, R., Lebeltel, O., Ripado, R., Girard, A., Dang, T., Maler, O.: SpaceEx: Scalable verification of hybrid systems. In: Computer Aided Verification (CAV). LNCS, Springer (2011)
14. Gurfil, P., Kholshevnikov, K.V.: Manifolds and metrics in the relative spacecraft motion problem. Journal of Guidance Control and Dynamics 29(4), 1004–1010 (2006)
15. Henzinger, T.A., Ho, P.H., Wong-Toi, H.: Hytech: a model checker for hybrid systems. Journal on Software Tools for Technology Transfer 1, 110–122 (1997)
16. Kaynar, D.K., Lynch, N., Segala, R., Vaandrager, F.: The Theory of Timed I/O Automata. Synthesis Lectures in Computer Science, Morgan & Claypool (2006)
17. Kurzhanskiy, A., Varaiya, P.: Ellipsoidal toolbox. In: 45th IEEE Conference on Decision and Control (CDC). pp. 1498–1503 (Dec 2006)
18. Lynch, N., Segala, R., Vaandrager, F.: Hybrid i/o automata. Inf. Comput. 185(1), 105–157 (2003)
19. Masur, H., Tabachnikov, S.: Chapter 13. Rational billiards and flat structures. In: Hasselblatt, B., Katok, A. (eds.) Handbook of Dynamical Systems, Handbook of Dynamical Systems, vol. 1, Part A, pp. 1015–1089. Elsevier Science (2002)
20. Ocampo, C.: Finite burn maneuver modeling for a generalized spacecraft trajectory design and optimization system. Annals of the New York Academy of Sciences 1017(1), 210–233 (2004)
21. Römgens, B.A., Mooij, E., Naeije, M.C.: Verified interval orbit propagation in satellite collision avoidance. In: Proc. AIAA Guidance, Navigation, and Control. Portland, OR (Aug 2011)
22. Vallado, D., McClain, W.: Fundamentals of astrodynamics and applications. Space technology library, Microcosm Press (2001)