arXiv:2505.23430v1 [math.DS] 29 May 2025

Locating Extremal Periodic Orbits for the Planar Circular Restricted Three Body Problem using Polynomial Sum-of-Squares Optimization

Vinay Sharma[1] and Sergei I. Chernyshenko

Department of Aeronautics, Imperial College London, Exhibition Road, London, SW7 2AZ, United Kingdom.

Abstract

With an increasing interest in the design of long and complex space missions, the search for orbits that require the least amount of fuel is of fundamental interest. This paper develops existing computational models for locating Unstable Periodic Orbits (UPOs) in polynomial dynamical systems using Sum-of-Squares (SOS) optimization technique and proposes a numerical framework to converge UPOs for the Planar Circular Restricted Three-Body Problem (PCR3BP) in astrodynamics. This is done by developing the polynomial SOS optimization technique with extension to systems with non-polynomial and Hamiltonian dynamics. First, we demonstrate and exploit the dependency of convergence of tight bounds on an observable of interest with varying scaling factors for large polynomial degrees. SOS optimization is then used to compute nonnegative polynomials, the minimization sublevel sets of which, approximately localise parts of the corresponding UPO. Improvements in current non-linear optimization techniques are suggested to compute a large number of points inside the relevant sublevel sets. Such points provide good initial conditions for UPO computations with existing algorithms. The distinguishing feature of such UPOs is that they optimize the long-time average of an input observable of interest which is a function of state variables. For the PCR3BP this means that such orbits in space can be traversed indefinitely in time without continuous fuel expenditure. As practical applications to space mission designs, we converge UPOs that minimise transmitted power required by satellites for the Earth-Moon system in a communication relay problem by minimizing the infinite-time average of sum of squares of distances of a satellite from Earth and the Moon.

Keywords: PCR3BP, Astrodynamics, SoS Optimization, Unstable Periodic Orbits

[1]Corresponding author: vinay.sharma@eng.ox.ac.uk

1 Introduction

1.1 Motivation and Objectives

Periodic orbits are specific trajectories that repeat over time, and they can have practical applications in space mission planning, such as maintaining stable orbits or enabling efficient transfers between different regions of space. In the classical three-body problem, each celestial body influences the motion of every other body in the system. Periodic solutions to the three-body problem act as building blocks to study the four-body problem and are a critical foundation for understanding the dynamics of the solar system at large. The gravitational force between two bodies is proportional to the product of their masses and inversely proportional to the square of the distance between them. For three or more bodies, this leads to a complex system of differential equations wherein predicting the long-term behaviour of bodies poses substantial challenges due to their inherent chaotic nature [16]. The sensitivity to initial conditions in chaotic systems leads to a loss of predictability over time, rendering accurate trajectory estimation a formidable task. The discovery of Unstable Periodic Orbits (UPOs) has significant advantages as they are dense in chaotic systems [6]. By applying control strategies to bring trajectories close to UPOs, it becomes possible to influence the system's behaviour [24]. In the context of astronavigation, UPOs, despite being unstable, present trajectories that can be traversed repeatedly without continuously expending propellant. This unique characteristic paves the way for the design of extended space missions to observe celestial bodies over vast distances and durations.

The main objective of this work is to provide a computational framework to converge UPOs for PCR3BP that are extremal in the sense that, in the $n-$dimensional state space $\boldsymbol{a} \in \mathbb{R}^n$, they optimize the infinite-time average of a physical quantity of interest $\Phi(\boldsymbol{a})$. We exploit a discovery made by Tobasco *et al.* [31] between so-called auxiliary function used to prove infinite-time averages on $\Phi(\boldsymbol{a})$ and the corresponding extremal trajectories, which are often UPOs. Precisely, if an auxiliary function produces nearly sharp bounds, then it can be used to construct a function whose sublevel sets localise extremal and near-extremal trajectories in state space. For polynomial ODE systems, Chernyshenko *et al.* [4] proposed a strategy of using auxiliary functions to prove sharp bounds on infinite-time average of $\Phi(\boldsymbol{a})$. This approach produces an optimization problem over the set of non-negative polynomials which is an NP-Hard problem [23] for modern day computers. However, a polynomial nonnegativity constraint can be strengthened into the condition that the polynomial admits a Sum-of-Squares (SOS) decomposition of lower-degree polynomials. The optimisation of auxiliary functions subject to SOS constraints can be translated into a Semi-Definite Program (SDP) [25, 26] — a standard type of convex optimisation problem. Although SOS constraints are only sufficient for non-negativity and are not necessary, except for polynomials as studied in [27], previous works [22] show that for most dynamical system of practical importance, they function exceptionally well.

The rest of the work is structured as follows. In section 2, we study the PCR3BP dynamical system in detail and motivate the search for UPOs of specific energy levels, for interplanetary space missions. In section 3, we review the technique of using auxiliary functions to prove sharp bounds on infinite-time averages, and as a technique proposed by Tobasco *et al.* [31], use these bounds to locate extremal UPOs. We review the SOS optimization technique and how the non-negativity of polynomial expressions can be reduced to the polynomials exhibiting SOS decomposition. In section 4, to extend the analysis to non-polynomial and Hamiltonian systems like PCR3BP, polynomial multipliers constructed using rational expressions and approximation of radical terms of the system dynamics, are proposed. Through an illustration with the Van der Pol oscillator system, we introduce two of the original contributions of this work, namely− the connection between scaling factors and convergence of shape bounds for large polynomial degrees; and the introduction of a novel computational framework of

nonlinear minimization to locate a large set of initial conditions, thus uncovering most parts of the target UPO. Section 5 discusses the implementation of the extended methodology to PCR3BP. Results are presented in section 6. In section 7, two newly discovered UPOs for the PCR3BP are presented, one of which minimises long-time power transmission by a satellite in a communication relay problem for the Earth-Moon system.

1.2 Problem Statement

Consider the situation shown in Figure 1. We have a satellite, P, of negligible mass, in the gravitational field of two massive bodies m_1 and m_2. Our goal is to find a periodic orbit on which the infinite-time average of a certain objective function ($\Phi(\boldsymbol{a})$) of space variables takes extremal value. Parts of such an orbit can also be used as a transfer orbit between an initial and a final orbit around the primaries.

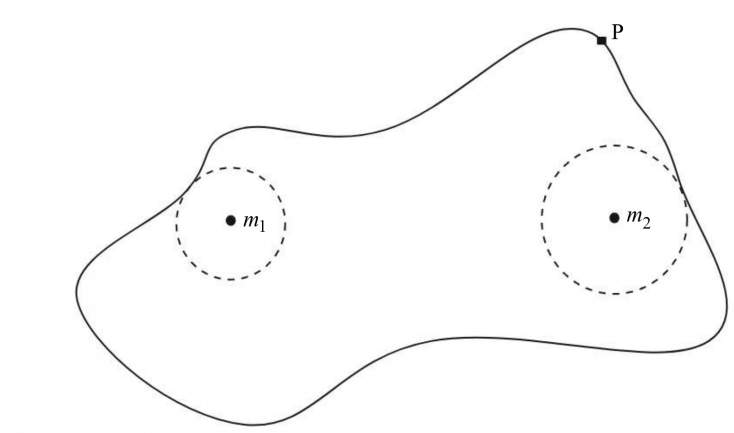


Fig. 1: Primaries m_1 and m_2 (•), test particle or a satellite (P), circular orbits in close vicinity of the primaries (- -) and target UPO to be searched for (—)

.

2 Overview of PCR3BP Dynamics

The three-body problem in astrodynamics is a complex and classical problem that involves studying the motion of three massive celestial bodies (usually stars, planets, or moons) under the influence of their mutual force of gravitational attraction. This force between the bodies lead to intricate and often chaotic trajectories [16, Section 3.10], making it challenging to predict their long-term behaviour accurately. Discovering a large number of UPOs for the three-body problem would open new possibilities of space-mission design [13].

The PCR3BP is a simplified problem obtained with some assumptions imposed on the general three-body problem. We confine the motion of three celestial bodies to a common plane ('Planar'). The two massive bodies, or the primaries, with mass m_1 and m_2 are assumed to be moving in circular orbits

around their common centre of mass ('Circular'). The third body is a test body of unit mass that moves under the gravitational influence but, is restricted to be too small to affect the motion of the primaries ('Restricted'). This problem finds applications in space missions involving small spacecraft or satellites orbiting around much larger celestial bodies. If we manoeuvre a satellite on a UPO by prescribing the correct spatial and velocity components, it will follow the path of the UPO indefinitely, without a need for continuous propellant expenditure. There exist heteroclinic and homoclinic connections [16, 9] between UPOs which can also be exploited for more complex space missions, see for example [15, 8].

We now look at some key expressions that lead to the formulation of this dynamical system. Let the mass parameter (μ) denote the mass ratio $m_2/(m_1+m_2)$. With the simplifications proposed above, we can normalize the distance between the primaries to be unity. Furthermore, in the rotating frame, normalized to rotate with unit angular velocity, the two bodies may be located on a horizontal axis at the points $(-\mu, 0)$ and $(1-\mu, 0)$. In the normalised state space $\boldsymbol{a} = [x, y, u, v]^T$, let the position and velocity of the satellite be denoted (x, y) and (u, v) respectively. The dynamical equation of the PCR3BP with simplifications proposed above takes the following form:

$$
\begin{aligned}
\frac{\mathrm{d}x}{\mathrm{dt}} &= u \\
\frac{\mathrm{d}y}{\mathrm{dt}} &= v \\
\frac{\mathrm{d}u}{\mathrm{dt}} &= 2v + x - \frac{(1-\mu)(x+\mu)}{\left(\sqrt{(x+\mu)^2+y^2}\right)^3} - \frac{\mu(x-1+\mu)}{\left(\sqrt{(x-1+\mu)^2+y^2}\right)^3} \\
\frac{\mathrm{d}v}{\mathrm{dt}} &= -2u + y - \frac{(1-\mu)y}{\left(\sqrt{(x+\mu)^2+y^2}\right)^3} - \frac{\mu y}{\left(\sqrt{(x-1+\mu)^2+y^2}\right)^3}
\end{aligned}
\tag{1}
$$

2.1 Energy Surface and Realms of Possible Motion

The equations of PCR3BP (1) are derived using the Lagrangian approach in the rotating frame. This problem also has a well-known Hamiltonian form [16], and the motion of the particle, P, obeys a conserved energy integral $E(\boldsymbol{a})$ given by (2). The community of scientists who study dynamical astronomy, often alternatively use the Jacobi integral $J(\boldsymbol{a}) = -2E(\boldsymbol{a})$ as the conserved quantity.

$$
E(\boldsymbol{a}) = \frac{1}{2}\left(u^2 + v^2\right) + \overline{U}(x, y), \tag{2}
$$

where $\overline{U}$ is known as the augmented or effective potential as is defined as

$$
\overline{U} = -\frac{1}{2}(x^2+y^2) - \frac{1-\mu}{\sqrt{(x+\mu)^2+y^2}} - \frac{\mu}{\sqrt{(x-1+\mu)^2+y^2}} \tag{3}
$$

To be able to design space missions by converging orbits for PCR3BP, it is critical that we review the concepts associated with energy surface. For a fixed mass parameter μ and energy level e, the equation $E(\boldsymbol{a}) = e$, describes a unique energy surface which is a three-dimensional set in the four-dimensional state space ($\boldsymbol{a}$). The energy surface provides a useful way to visualize the allowed regions of motion for the test particle, called "realms" of possible motion. The region of space in $x-y$ subspace where the motion of the particle is prohibited for some prescribed energy level is called the "forbidden" realm. Reals of possible motion are also called Hill's region, given mathematically by the projection of $E(\boldsymbol{a})$

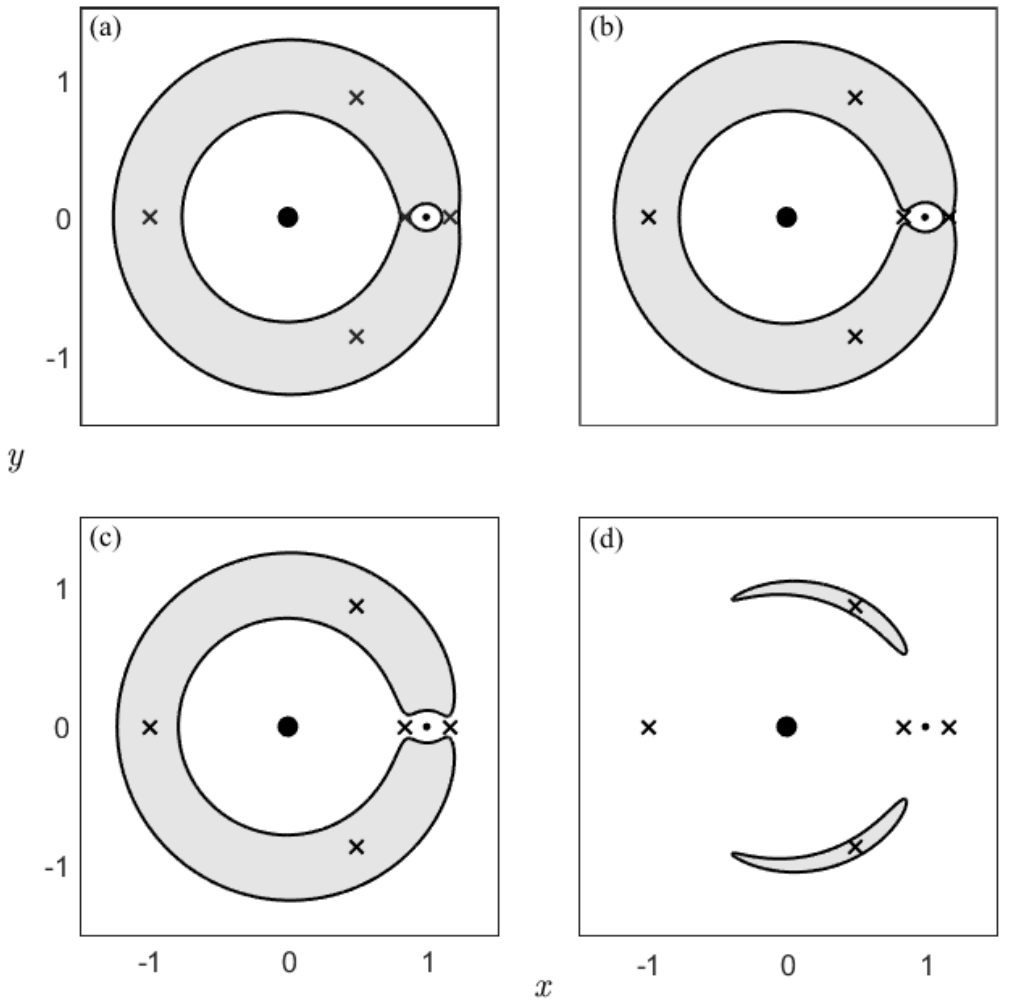


Fig. 2: Four configurations of the Hill's regions (areas in white) and forbidden realms (area in grey) in the x-y subspace for Earth (●) and Moon (•). Zero velocity curves (—), where the kinetic energy of the third body vanishes. (a) $e_1 < E$, (b) $e_1 < E < e_2$, (c) $e_2 < E < e_3$, (d) $e_3 < E < E_4$. Lagrangian points (×).

onto the $x - y$ subspace as:

$$C(\mu, e) = \{(x, y) \,|\, \overline{U} \leq e\} \tag{4}$$

Note that a choice of e in $E(\boldsymbol{a}) = e$ prescribes an equivalent height of the effective potential energy function $\overline{U}$. Let us now take a look at the four fundamental classifications (see Figure 2) of the Hill's region depending on the value of e.

- $E < e_1$: There is no transit motion possible between the two permissible realms. The particle P is trapped where it can either have bounded motion or collision with one of the primaries.
- $e_1 < E < e_2$: The "neck" between the primaries open up. The particle can now have transition motion around the two primaries.
- $e_2 < E < e_3$: Another neck opens up where the particle can now move in the realm that extends to infinity.
- $e_3 < E < e_4$: The forbidden realm becomes limited in space and the particle can move towards infinity from the vicinity of either primaries.

For $E > e_4$, the forbidden realms do not exist and the particle is free to move in the entirety of $x - y$ subspace. The dynamical system (1) has five Lagrangian points [29] which are points of equilibria (see Figure 2) where $\frac{\partial \overline{U}}{\partial x} = \frac{\partial \overline{U}}{\partial y} = 0$. These are important topographical points in placing instruments for unobstructed observation of primaries and act as passageways for travel beyond low-energy orbits, discussed more in [16]. It is important to note that the numerical framework we are developing has no reservations about the choice of either μ or e. However, for the demonstration of one of many possible applications, we will search for UPOs for the Earth-Moon system ($\mu = 0.01250$). Furthermore, our aim of designing a transfer orbit or one that optimizes the infinite-time average of an observable of interest for a pair of celestial bodies, the second case ($e_1 < E < e_2$) offers the most interesting and practical grounds.

3 Localising Extremal UPOs using Auxiliary Functions

Our starting point is an autonomous dynamical system governed by the ODE

$$\frac{\mathrm{d}\boldsymbol{a}}{\mathrm{d}t} = \boldsymbol{f}(\boldsymbol{a}), \quad \boldsymbol{a} \in \mathbb{R}^n. \tag{5}$$

Let the observable of interest for the system represented by (5) be $\Phi(\boldsymbol{a})$. We assume that this function is polynomial. Let us assume that all the trajectories of (5) eventually lie inside a compact set Ω, even if they leave this set transiently, for all initial conditions $\boldsymbol{a}_0 \in \mathbb{R}^n$. For a trajectory starting from $\boldsymbol{a}_0$, the infinite-time average of $\Phi(\boldsymbol{a})$ is defined as

$$\overline{\Phi}\left(\boldsymbol{a}_0\right) := \lim_{\mathrm{T}\to\infty} \frac{1}{\mathrm{T}} \int_0^{\mathrm{T}} \Phi\left(\boldsymbol{a}\left(t; \boldsymbol{a}_0\right)\right) \mathrm{d}t. \tag{6}$$

There could be several ways of topologically defining the compact set Ω and infinite-time average, as in [31], but require no modification to the theory that follows. We are interested in the maximal (or minimal) value of $\overline{\Phi}(\boldsymbol{a})$ over all trajectories,

$$\overline{\Phi}^* := \max_{\substack{\boldsymbol{a_0}\in\Omega \\ ||\boldsymbol{a}(t;\boldsymbol{a_0})||<\infty \forall t}} \overline{\Phi}(\boldsymbol{a_0}) \tag{7}$$

as well as the initial conditions and corresponding trajectories which achieve it. Let the auxiliary function of state variables be denoted by $V(\boldsymbol{a}) \in C^1(\Omega)$. The time derivative of the globally continuous, differentiable and bounded function $V(\boldsymbol{a}(t))$ (using the chain rule of differentiation) is:

$$\frac{\mathrm{d}V\boldsymbol{a}(t)}{\mathrm{d}t} = \nabla V(\boldsymbol{a}(t)) \cdot \boldsymbol{f}(\boldsymbol{a}(t)) \tag{8}$$

There is a key feature of the RHS of (7) and (9). The variable t is an argument in $\boldsymbol{a}(t)$. Hence, for a fixed time t, there exists infinite possibilities for optimal ${\boldsymbol{a}_{\boldsymbol{0}}}^*$ such that $\overline{\Phi}(\boldsymbol{a}_{\boldsymbol{0}}^*) = \overline{\Phi}^*$ and $\boldsymbol{a}$ for which the derivative of the auxiliary function $V(\boldsymbol{a}(t))$ takes the form in (9). We can easily calculate sharp lower (or upper) bounds on $\bar{\Phi}^*$ using an approach originally proposed by Chernyshenko *et al.* [4]. For its implementation, let us derive the infinite-time average of the time-derivative of auxiliary functions. Using (9), we have:

$$\begin{aligned}\overline{\frac{\mathrm{d}V\left(\boldsymbol{a}\left(t;\boldsymbol{a}_0\right)\right)}{\mathrm{d}t}} &= \overline{\boldsymbol{f}\left(\boldsymbol{a}\left(t;\boldsymbol{a}_0\right)\right)\cdot\nabla V\left(\boldsymbol{a}\left(t;\boldsymbol{a}_0\right)\right)}\\ &= \lim_{\mathrm{T}\to\infty}\frac{V\left(\boldsymbol{a}\left(\mathrm{T};\boldsymbol{a}_0\right)\right)-V\left(\boldsymbol{a}_0\right)}{\mathrm{T}}\\ &= 0\end{aligned} \tag{9}$$

Let us say there exists a constant L such that $\Phi(\boldsymbol{a})+\boldsymbol{f}(\boldsymbol{a})\cdot\nabla V(\boldsymbol{a})\geq L$ for any $\boldsymbol{a}\in\Omega$. Then, averaging this inequality along the trajectory $\boldsymbol{a}\left(t;\boldsymbol{a}_0\right)$ yields $\overline{\Phi}\left(\boldsymbol{a}_0\right)\geq L$ for any $\boldsymbol{a}_0\in\Omega$. In accordance with (7), we have $\overline{\Phi}^*\geq L$. We can obtain the optimal lower bound (L) on $\overline{\Phi}^*$ by optimising over the choice of $V(\boldsymbol{a})$ as following:

$$\overline{\Phi}^*\geq\sup_{\substack{V(\boldsymbol{a})\in C^1(\Omega)\\ L\in\mathbb{R}}}\left\{L\mid\Phi(\boldsymbol{a})+\boldsymbol{f}(\boldsymbol{a})\cdot\nabla V(\boldsymbol{a})-L\geq 0\text{ on }\Omega\right\} \tag{10}$$

Tobasco *et al.* [31] proved that when Ω is compact, (10) under all current assumptions on the trajectories of (5) is actually an equality. Hence, the extremal infinite-time averages are *exactly* characterised by:

$$\overline{\Phi}^*=\sup_{\substack{V(\boldsymbol{a})\in C^1(\Omega)\\ L\in\mathbb{R}}}\left\{L\mid\Phi(\boldsymbol{a})+\boldsymbol{f}(\boldsymbol{a})\cdot\nabla V(\boldsymbol{a})-L\geq 0\text{ on }\Omega\right\}. \tag{11}$$

As discussed in section 1.1, solving for RHS of (11) is an NP-Hard problem. We can replace the non-negativity of (11) by its condition of exhibiting an SOS decomposition. The problem of proving the best bound over the space $\Pi_{n,d}$ can be simplified to the following computationally tractable SOS problem by optimising:

$$\overline{\Phi}^*\geq\sup_{\substack{V(\boldsymbol{a})\in\Pi_{n,d}\\ L\in\mathbb{R}}}\left\{L\mid\Phi(\boldsymbol{a})+\boldsymbol{f}(\boldsymbol{a})\cdot\nabla V(\boldsymbol{a})-L\in\Sigma_{n,d}\right\} \tag{12}$$

where the coefficients of $V(\boldsymbol{a})\in\Pi_n(\boldsymbol{a})$ are decision variables. Similar problems using auxiliary functions are further studied in [4, 7, 10]. A crucial aspect of (12) emerges – we can optimize auxiliary functions to yield robust upper bounds on $\overline{\Phi}^*$ without needing to solve the ODE (2.1). Especially in the case of highly nonlinear or chaotic systems, attaining nearly accurate bounds might prove easier than directly calculating $\overline{\Phi}^*$ through extensive numerical simulations. Moreover, for systems with complex dynamics, optimizing a fully general $V(\boldsymbol{a})$ might not be straightforward. Nevertheless, we can practically obtain excellent bounds by considering subsets of auxiliary functions that we can optimize numerically. We see this demonstrated clearly in [10, 12], where Goluskin *et al.* constructed nearly optimal polynomial auxiliary functions for the Lorenz system and the Kuramoto-Sivashinsky equation using SOS optimization.

As we explore the concept further, we discover that auxiliary functions not only provide sharp bounds on $\overline{\Phi}$ but also assist in localizing the associated extremal trajectories in state space. Imagine

that a fully optimal auxiliary function V^*, resulting in a bound L exactly matching $\overline{\Phi}^*$, is available. In such a case, the extremal trajectory $\boldsymbol{a}(t)$ satisfies the condition [7, 10]:

$$\overline{\Phi(\boldsymbol{a}) + \boldsymbol{f}(\boldsymbol{a}) \cdot \nabla V^*(\boldsymbol{a}) - L} = 0. \tag{13}$$

Here, $L = \overline{\Phi}^*$. Since the averaged quantity is nonnegative, it must essentially vanish pointwise in time, signifying that the extremal trajectory resides within the set:

$$\mathcal{S}_0 := \{\boldsymbol{a} \in \Omega \mid \Phi(\boldsymbol{a}) + \boldsymbol{f}(\boldsymbol{a}) \cdot \nabla V^*(\boldsymbol{a}) - L = 0\} \tag{14}$$

While not all points within $\mathcal{S}_0$ necessarily belong to the extremal trajectory, they do offer valuable guidance for locating it. However, it is worth noting that the existence of an optimal auxiliary function is not guaranteed due to the nonattainment of the supremum in (12). Even if such an optimal function exists, computing it might be a challenge. Nevertheless, equation (12) assures us that we can find a δ-suboptimal auxiliary function for any $\delta > 0$. Specifically, there exists a $V \in \Pi_n$ that provides a bound L satisfying $\overline{\Phi} \leq L \leq \overline{\Phi} + \delta$. Although numerically constructing such a function for high-dimensional systems could demand extensive computational resources, it is always feasible. As detailed in [31], even a suboptimal V computed numerically can help us roughly locate the extremal trajectory, though the results may be weaker compared to an optimal V.

To elaborate further, for any pair (L, V), where L forms a δ-suboptimal bound on $\overline{\Phi}^*$ and V represents the corresponding δ-suboptimal polynomial auxiliary function, let us define the nonnegative function $D_{L,V}(\boldsymbol{a})$ as follows:

$$D_{L,V}(\boldsymbol{a}) := \Phi(\boldsymbol{a}) + \boldsymbol{f}(\boldsymbol{a}) \cdot \nabla V(\boldsymbol{a}) - L \tag{15}$$

Now, consider the set $\mathcal{S}\varepsilon$ containing points where $D_{L,V}(\boldsymbol{a})$ is not greater than a chosen $\varepsilon > 0$:

$$\mathcal{S}_\varepsilon = \{\boldsymbol{a} \in \Omega \mid D_{L,V}(\boldsymbol{a}) \leq \varepsilon\} \tag{16}$$

When we possess a δ-suboptimal $V(\boldsymbol{a})$, we can be certain that the extremal trajectory lies within $\mathcal{S}_\varepsilon$ for a duration determined by the specific values of ε and δ. For any δ-suboptimal V, the extremal trajectory is guaranteed to lie in $\mathcal{S}_\varepsilon$ for a fraction of time determined by the values of ε and δ [10, Section 3]. The points of minimization of δ-suboptimal D lie in sub-level set $\mathcal{S}_\varepsilon$ and can be used as initial conditions to converge the corresponding extremal trajectories, which are often UPOs.

3.1 Non-negativity on Semi-algebraic Sets

It is often of physical significance, in the study of some dynamical systems, to prove the non-negativity of certain polynomials inside a region of interest. Such a region could be a local attractor, for instance. Let us define a semialgebraic set S, by a polynomial equation and a finite number of polynomial inequalities such that:

$$S = \{\boldsymbol{a} \in \mathbb{R}^n \mid G_1(\boldsymbol{a}) \geq 0, \ldots, G_m(\boldsymbol{a}) \geq 0, H(\boldsymbol{a}) = 0\} \tag{17}$$

We can establish the certification of nonnegativity of polynomial $P(\boldsymbol{a})$ inside the set S, using the following set of relations:

$$\begin{aligned} &D(\boldsymbol{a}) = P(\boldsymbol{a}) + \sigma(\boldsymbol{a})H(\boldsymbol{a}) - \sum_{j=1}^{m} s_j(\boldsymbol{a})G_j(\boldsymbol{a}) \geq 0 \\ &s_j(\boldsymbol{a}) \geq 0 \quad \text{for} \quad i = 1, \ldots, m \\ &\sigma(\boldsymbol{a}) \in \mathbb{R}^n \end{aligned} \tag{18}$$

A weaker but sufficient formulation of (18) in terms of SOS constraints is discussed below.

Theorem 2.1 (Generalised S-procedure [3, Section 3.2]) A polynomial $P(\boldsymbol{a})$ is nonnegative on the set S, defined as in (17), if there exist polynomials $\sigma(\boldsymbol{a}) \in \Pi_n$ and $s_1(\boldsymbol{a}), \ldots, s_m(\boldsymbol{a}) \in \Sigma_n$ such that:

$$D(\boldsymbol{a}) = P(\boldsymbol{a}) + \sigma(\boldsymbol{a})H(\boldsymbol{a}) - \sum_{j=1}^{m} s_j(\boldsymbol{a})G_j(\boldsymbol{a}) \in \Sigma_n \tag{19}$$

Thus, non-negativity constraints on semialgebraic sets can also be handled computationally with SOS relaxations. Finally, it should be noted that the Generalised S-procedure can be derived more rigorously from first principles by considering a theorem from real algebraic geometry known as Stengle's Positivstellensatz. Henceforth, we will reduce the problem of proving the non-negativity of polynomial functions with a sufficient condition of it exhibiting an SOS decomposition.

4 Extending Methodology to Non-polynomial & Hamiltonian Systems

So far, we reviewed a strategy of how we can, for a polynomial dynamical system (5), prove sharp extremal bounds on infinite-time average of a quantity of interest ($\Phi(\boldsymbol{a})$) and converge corresponding UPOs. In principle, such an analysis can also be extended to the class of systems with non-polynomial and Hamiltonian dynamics. One obvious way is to approximate the non-polynomial parts of RHS of (5) using polynomial functions, as demonstrated in [28]. For an $n-$dimensional non-polynomial system, it is also intuitive to derive n additional scalar ODEs which represent time derivatives of non-polynomial terms [19, Section 7.3]. However, with their application to the specific problem of finding UPOs for the PCR3BP, although, these strategies do not explicitly declare reservations on the choices of the function $\Phi(\boldsymbol{a})$, the only known UPOs produced by these works [19, Section 7.4.4] have relied on an "artificial" $\Phi(\boldsymbol{a})-$one that relies on *a priori* knowledge of an existing UPO. The problem then becomes of finding UPOs lying at the same energy level as that of the known one. This is a limitation for discovering optimal UPOs for space mission designs with the demand for more practical applications of $\Phi(\boldsymbol{a})$. This section aims to address this limitation.

Recall, for polynomial dynamics, if there exists a constant L such that $\Phi(\boldsymbol{a}) + \boldsymbol{f}(\boldsymbol{a}) \cdot \nabla V(\boldsymbol{a}) \geq L$ for any $\boldsymbol{a} \in \Omega$, we can obtain the optimal lower bound (L) on $\overline{\Phi}^*$ by optimising over the choice of $V(\boldsymbol{a})$ as following:

$$\overline{\Phi}^* \geq \sup_{\substack{V(\boldsymbol{a}) \in \Pi_n \\ L \in \mathbb{R}}} \{L \mid \Phi(\boldsymbol{a}) + \boldsymbol{f}(\boldsymbol{a}) \cdot \nabla V(\boldsymbol{a}) - L \geq 0 \text{ on } \Omega\}$$

Let us now consider the case when $\boldsymbol{f(a)}$ represents a class of non-polynomial and Hamiltonian dynamical systems. Let $R(\boldsymbol{a})$ be a positive-definite polynomial multiplier formed using denominators of rational parts and approximation of radical parts of $\boldsymbol{f(a)}$ and let $\boldsymbol{F(a)}$ represent the system of dynamical

equations obtained by multiplying $R(\boldsymbol{a})$ to polynomial parts of $\boldsymbol{f(a)}$. In consistency with the methodology developed for polynomial functions, we can prove a sharp lower bound (L) on the infinite-time average of $\Phi(a)$ by multiplying $R(\boldsymbol{a})$ to both sides of the inequality above. This optimization problem can be formulated for all trajectories that lie inside the set S, defined in (17), in accordance with generalised $S-$procedure (see Section 3.1). Simplified to a computationally tractable formulation of SOS approximation, we can reformulate the optimization problem as:

$$\overline{\Phi}^{*} \geq \sup_{\substack{V(\boldsymbol{a})\in\Pi_n \\ \sigma(\boldsymbol{a})\in\Pi_n \\ s_1,\ldots,s_m\in\Pi_n \\ L\in\mathbb{R}}} \{L \mid D(\boldsymbol{a}) \in \Sigma_n \text{ and } s_j(\boldsymbol{a}) \in \Sigma_n \quad \text{for } j = 1,\ldots,m\} \tag{20}$$

where,

$$D(\boldsymbol{a}) = \boldsymbol{F}(\boldsymbol{a}) \cdot \nabla V(\boldsymbol{a}) + \big(\Phi(\boldsymbol{a}) - L\big)R(\boldsymbol{a}) + \sigma(\boldsymbol{a})H(\boldsymbol{a}) - \sum_{j=1}^{m} s_j(\boldsymbol{a})G_j(\boldsymbol{a}) \tag{21}$$

The equation 21 (used in conjunction with (4.1)) is the main highlight of this section. The introduction of the positive definite polynomial multipliers $R(\boldsymbol{a})$ is a key improvement from previous approximation strategies in the sense that it retains rational parts of $\boldsymbol{f(a)}$ and only approximates the radical parts in the formulation of $\boldsymbol{F(a)}$.

4.1 Illustrative Example: Van der Pol Oscillator

Let us now demonstrate the application of SOS optimization technique for inequalities of the form (11) using Van der Pol oscillator. The dynamical equation for this system is given by:

$$\begin{aligned} \frac{\mathrm{d}x}{\mathrm{d}t} &= y \\ \frac{\mathrm{d}y}{\mathrm{d}t} &= \mu(1-x^2)y - x \end{aligned} \tag{22}$$

where μ is the damping factor of the system. Note that, although, the RHS of (12) is polynomial, studying this system with extended methodology serves a critical purpose. It is obvious that the success of converging any UPO for non-polynomial dynamical systems is greatly dependent on the choice of $R(\boldsymbol{a})$ used for approximation. Better approximation often calls for the choice of high degrees of $R(\boldsymbol{a})$. However, SOS optimization problems in YALMIP for high polynomial degrees often become ill-conditioned [21]. This is due to the corresponding SDP problems growing rapidly causing non-convergence. We know from previous works of Fantuzzi *et al.* [7], the sharp bound on the infinite-time average of $\Phi(\boldsymbol{a}) = x^2 + y^2$ and the corresponding UPO on which it takes extremal values. This allows us to implement the extended analysis with randomly chosen positive definite polynomials $R(\boldsymbol{a})$ of increasing degrees to test the robustness of YALMIP in proving a sharp bound. As we will see, this study will lead us to a remarkable connection between the scaling factors employed in the optimization problem and the convergence of a sharp bound in YALMIP.

4.2 Numerical Implementation & Results

The numerical procedure of converging UPOs as detailed in this work is relatively straightforward and can be broadly summarized in the following steps:

1. For an identified observable of interest $\Phi(\boldsymbol{a})$, solve the optimization problem (11) using concepts of SOS relaxation as discussed in 3. Choose a minimum degree of the auxiliary function $V(\boldsymbol{a})$ such that the overall degree of the polynomial to be optimized, i.e., degree of $\Phi(\boldsymbol{a}) + \boldsymbol{f}(\boldsymbol{a}) \cdot \nabla V(\boldsymbol{a})$ is even.
2. Increase the degree of the auxiliary and solve the SOS optimization problem again.
3. Repeat steps 1 and 2 until the parameter for extremal lower bound L (or upper bound U) reaches convergence.
4. Using any suitable convex nonlinear minimization algorithm, minimize the polynomial $D(\boldsymbol{a})$ as in (15) to locate points of minimization. Such points lie on the corresponding UPO and can be used as initial conditions for integrating the UPO forward in time.

Two remarks on the implementation of the procedure highlighted are in order. First, we may be interested in converging UPOs which are locally extremal, as we will see with application to PCR3BP in section 5. In such cases, we must optimize on the polynomial with the $S-$procedure terms (20) in steps 1 to 4. Secondly, the implementation of step 2, i.e., solving the SOS optimization problem with increasing degree of auxiliary function $V(\boldsymbol{a})$ is often challenging for high degrees where the numerical errors grow out of bounds and infeasibility of the problem is reported by SOS solvers. The first instance of such a numerical failure for Van der Oscillator takes place when the degree of $V(\boldsymbol{a})$ is increased to 12 and the overall degree of polynomial $D(\boldsymbol{a})$ is 14, marked in the Table 1 with †. To ensure convergence, it is a common practice [19, 17, 18] to scale the system dynamics to within a unit box such that:

$$\boldsymbol{a_s} = \boldsymbol{a} \cdot \boldsymbol{A_s} + \boldsymbol{B} \tag{23}$$

where the elements of the n-dimensional row vector $\boldsymbol{A_s}$ gives the maximum span of all trajectories of (5) along each component of $\boldsymbol{a} \in \mathbb{R}^n$. The vector $\boldsymbol{B}$ is used to position the n-dimensional unit box at the origin $\mathbf{O}$.

It is important to note that all row entries listed subsequent to the one marked '†' require system scaling for convergence of a bound. However, in practise, for yet higher degrees of the polynomial $D(\boldsymbol{a})$, the problem often turns numerically infeasible. We may still desire to be able to increase the degree even further, either because the parameter L (or U) is not converged or the polynomial $R(\boldsymbol{a})$ is required to attain higher degrees for a better approximation.

This brings us to the first original contribution of this work, specifically that, there is a connection between the scaling factors $\boldsymbol{A_s}$ & $\boldsymbol{B}$ and convergence of a bound in YALMIP. Scaling a system down to different n-dimensional boxes ensures convergence for practically the highest desired degree of polynomial $D(\boldsymbol{a})$ required to exhibit SOS decomposition. We demonstrate this with Van der Oscillator where the degree of polynomial $D(\boldsymbol{a})$ is raised to 30. Note that, all row entries marked with '‡' employ a newer set of scaling factors than the previous row entries.

The convergence of a sharp bound brings us to Step 4, where we use a convex nonlinear minimization algorithm like MATLAB's built-in function 'fmincon' to locate points of local minima of $D(\boldsymbol{a})$ that can be used as initial conditions for converging a UPO. A set of 100 random initial conditions in $[-1, 1]$ for the minimization algorithm produces good initial conditions in (x, y) shown in Figure 3(a). Previous works [19, Section 6.3] have suggested choosing 5000 random initial conditions in $[-1, 1]$ to uncover most parts of the UPO which is computationally very expensive.

The second original contribution of this work is an improved numerical framework for locating a large set of initial conditions of a UPO. In the minimization of $D(\boldsymbol{a})$ each starting point travels through several local minima but only terminates at the final point when tolerances employed (discussed briefly in section 6) in the minimization scheme are satisfied. These local minima can also be candidates for good initial conditions for UPO convergence. The result of implementing this technique is shown in

deg($\boldsymbol{f}$)	deg(R)	deg(V)	deg(D)	Upper Bound (U)
3	0	8	10	4.297823887
3	0	10	12	4.122856510
3	0	12	14†	4.118793409
3	0	14	16	4.118765932
3	0	16	18	4.118762625
3	2	16	20‡	4.118802989
3	2	18	22	4.118784082
3	2	20	24‡	4.118756152
3	4	16	22‡	4.118752527
3	4	18	24	4.118749126
3	4	20	26	4.118756014
3	6	14	22‡	4.118754299
3	6	16	24‡	4.118753924
3	6	18	26‡	4.118753649
3	8	16	26	4.118753911
3	8	18	28	4.118752899
3	8	20	30	4.118749996
3	10	14	26	4.118756632
3	10	16	28‡	4.118754368
3	10	18	30	4.118737136

Table 1: Upper Bound (U) computed on the infinite-time average of $\Phi(\boldsymbol{a}) = x^2 + y^2$. The degree of auxiliary function (V) is increased for successive increases in the degree of polynomial multiplier (R) to converge sharp bounds. The row marked (†) indicates the degree of $D(\boldsymbol{a})$ at and beyond which scaling factors are required for convergence. Rows marked (‡) indicate employment of a newer set of scaling factors from previous rows for successful convergence.

Figure 3(b) where over 2700 suitable initial conditions were obtained from just 100 randomly distributed points. The code is available to the reader for use on request.

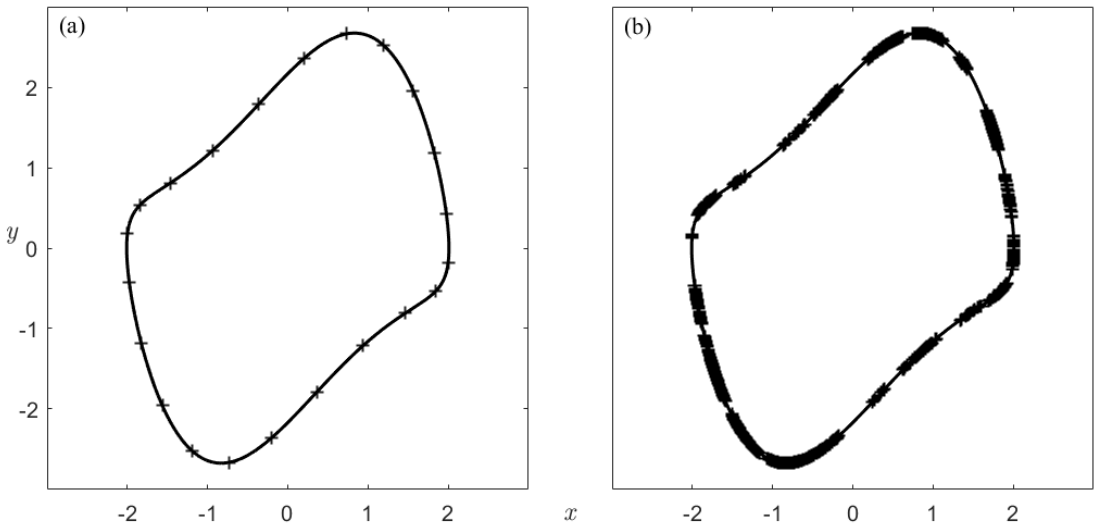


Fig. 3: Converged extremal UPO for $\Phi(\boldsymbol{a}) = x^2 + y^2$ (—). (a) Points of local minimizers (+) of $D(\boldsymbol{a})$ obtained using 'fmincon'. (b) Point of local minimizers (+) of $D(\boldsymbol{a})$ obtained with improvements in existing non-linear optimization framework of 'fmincon' to uncover large parts of UPO.

5 Application to PCR3BP

Thus far we have learnt the concept of converging UPOs for both− polynomial and non-polynomial dynamical systems, using auxiliary functions and SOS methodology. Additionally, we proposed two key concepts that augment existing numerical frameworks of UPO convergence employed in similar studies. First, discovering the relationship between scaling factors ($\boldsymbol{A_s}$ and $\boldsymbol{B}$) and convergence of a bound enables us to implement the SOS methodology to polynomials of high degrees. We will see in this section how this allows us to replace the non-polynomial parts of the PCR3BP system with approximated polynomials of high degrees. Secondly, our improvements in the existing numerical framework of non-linear minimization allow us to locate a large section of UPOs efficiently. Let us revisit the PCR3BP dynamical system (1). In our previous work [28], we approximated the reciprocal of radical expressions in the above set of equations with polynomial functions. Even for a range of degrees of such polynomials employed, the method showed limited success in the sense that *a priori* knowledge of an existing UPO is necessary to converge UPOs in its vicinity. Using rational functions for approximation rather than polynomials is a motivating argument. Here, we will exploit the structure of (1) in a way that uses both these approximation techniques. We can approximate the RHS as:

$$\begin{aligned} \frac{\mathrm{d}x}{\mathrm{dt}} &= u \\ \frac{\mathrm{d}y}{\mathrm{dt}} &= v \end{aligned} \tag{24}$$

$$\frac{\mathrm{d}u}{\mathrm{dt}} = 2v + x - \frac{(1-\mu)(x+\mu)}{\big((x+\mu)^2+y^2\big)} \cdot \frac{Q(z_1)}{P(z_1)} - \frac{\mu(x-1+\mu)}{\big((x-1+\mu)^2+y^2\big)} \cdot \frac{Q(z_2)}{P(z_2)}$$
$$\frac{\mathrm{d}v}{\mathrm{dt}} = -2u + y - \frac{(1-\mu)y}{\big((x+\mu)^2+y^2\big)} \cdot \frac{Q(z_1)}{P(z_1)} - \frac{\mu y}{\big((x-1+\mu)^2+y^2\big)} \cdot \frac{Q(z_2)}{P(z_2)}$$

where the rational function $\frac{P}{Q}$ is used to approximate the square-root terms inside the region of interest (as defined later in (27)) and $z_1 = \sqrt{(x+\mu)^2+y^2}, z_2 = \sqrt{(x-1+\mu)^2+y^2}$. Notice that with $R(\boldsymbol{a}) = \big((x+\mu)^2+y^2\big)\big((x-1+\mu)^2+y^2\big)P(z_1)P(z_2)$, $\boldsymbol{F(a)}$ takes the form:

$$\begin{aligned}
\boldsymbol{F}_1(\boldsymbol{a}) &= u \cdot \big((x+\mu)^2+y^2\big)\big((x-1+\mu)^2+y^2\big)P(z_1)P(z_2) \\
\boldsymbol{F}_2(\boldsymbol{a}) &= v \cdot \big((x+\mu)^2+y^2\big)\big((x-1+\mu)^2+y^2\big)P(z_1)P(z_2) \\
\boldsymbol{F}_3(\boldsymbol{a}) &= (2v+x) \cdot \big((x+\mu)^2+y^2\big)\big((x-1+\mu)^2+y^2\big)P(z_1)P(z_2)\cdots \\
&\quad - (1-\mu)(x+\mu)\big((x-1+\mu)^2+y^2\big)P(z_2)Q(z_1) - \mu(x-1+\mu)\cdot P(z_1)Q(z_2)\big((x+\mu)^2+y^2\big) \\
\boldsymbol{F}_4(\boldsymbol{a}) &= (-2u+y) \cdot \big((x+\mu)^2+y^2\big)\big((x-1+\mu)^2+y^2\big)P(z_1)P(z_2)\cdots \\
&\quad - (1-\mu)y \cdot \big((x-1+\mu)^2+y^2\big)P(z_2)Q(z_1) - \mu y \cdot P(z_1)Q(z_2)\big((x+\mu)^2+y^2\big)
\end{aligned} \tag{25}$$

It is an important stage for us to recall that the system we are dealing with is a Hamiltonian. This also sets our context for a search for UPOs that lie on a given energy level given by (1). For an energy level e, the first constraint $(E(\boldsymbol{a}) - e = 0)$ can be represented as polynomial equality using the technique implemented above. This formulates the first S-procedure constraint for bounding the region of interest is given by:

$$H(\boldsymbol{a}) := \big(u^2+v^2-x^2-y^2-2e\big)\,P(z_1)P(z_2) - 2\big(1-\mu\big)P(z_2)Q(z_1) - 2\mu P(z_1)Q(z_2) = 0 \tag{26}$$

Our choice of e defines one of the Hill-region configurations discussed in section 2.1. The region of interest is further bounded to lie inside a large circle in the $x-y$ plane enclosing most of the interior realm of the Hill's region. Let us also eliminate small areas around the primaries as the centre of the primaries are regions of singularities in $x-y$ subspace. Additional to the equality constraint of (26), the inequality constraints that define the region of interest are given by:

$$\begin{aligned}
G_1(\boldsymbol{a}) &:= (x+\mu)^2 + y^2 - R_E^2 \geq 0 \\
G_2(\boldsymbol{a}) &:= (x-1+\mu)^2 + y^2 - R_M^2 \geq 0 \\
G_3(\boldsymbol{a}) &:= R_G^2 - x^2 - y^2 \geq 0
\end{aligned} \tag{27}$$

Hence, we can employ redefined functional dynamics $\boldsymbol{F(a)}$ and optimise the polynomial $D(\boldsymbol{a})$ as in (21) over auxiliary functions to prove sharp bounds for a quantity of interest $\Phi(\boldsymbol{a})$.

6 Results

We discovered two new, closely lying UPOs for the Earth-Moon system at the energy level $e = -1.589070$ as plotted in Figure 5 and Figure 6. These two UPOs provide testimony to the fact the UPOs lie close in the $(n-1)$-dimensional set of Hamiltonian. The "petal" structure around Earth is common in both the converged UPOs however their behaviours around the moon vary in the number

of revolutions the UPOs take around it. $\Phi(\boldsymbol{a})$ is taken to be the sum of squares of distances of the satellite from Earth and the Moon.

$$\Phi(\boldsymbol{a}) = (x+\mu)^2 + 2y^2 + (x+\mu-1)^2 \tag{28}$$

The chosen expression for $\Phi(\boldsymbol{a})$ has the following significance. Effective communication in deep space is a complex domain that spans various engineering fields and entails a range of challenges. One of these challenges involves the use of communication relays [30]. A part of communication relay design is choosing the orbit for the relay. One approach is to select a few known orbits and then estimate their efficiency as the orbit of the relay, see for example [2]. However, this might lead to overlooking non-typical orbits that might be the best for the purpose. For a certified signal strength at the receiver, the power required for transmission, *ceteris paribus*, is directly proportional to the square of distance of the satellite from a celestial body. For the system in hand, total transmission power, hence, is directly proportional to the sum of squares of the distance of the satellite from Earth and the Moon. A smaller average of $\Phi(\boldsymbol{a})$ would result in lesser total power required for transmission. It is for these reasons that UPOs on which the infinite-time average of $\Phi(\boldsymbol{a})$ is minimum are desired.

Function	Degrees	Parameters	Values
$V(\boldsymbol{a})$	10	R_G	1.13
$s_1(\boldsymbol{a})$, $s_2(\boldsymbol{a})$, $s_3(\boldsymbol{a})$	10	R_G	0.01
$\boldsymbol{F}(\boldsymbol{a})$	13	R_G	0.01
$\sigma(\boldsymbol{a})$	10	μ	0.01250
$D(\boldsymbol{a})$	22	e	-1.58907

Table 2: Degrees of functions and parameter values for UPO computation in PCR3BP.

The choice of $e = -1.589070$ produces the Hills configuration as shown in Fig. 4. $S-$procedure constraints used in setting up bound computations are also shown. Note that for the rational approximations functions P and Q that lead to the polynomial formulation of $\boldsymbol{F}(\boldsymbol{a})$ and $H(\boldsymbol{a})$, we used the robust least-absolute residual method algorithm of the MATLAB's 'Curve fitting toolbox'. The degrees of auxiliary function $V(\boldsymbol{a})$, $S-$procedure multipliers $s_1(\boldsymbol{a}), s_2(\boldsymbol{a}), s_3(\boldsymbol{a})$ and $\sigma(\boldsymbol{a})$ are tabulated in Table 2. The obtained polynomial $D(\boldsymbol{a})$ is minimized using MATLAB's built-in function 'fmincon' subject to constraints used in $S-$procedure. The 'OptimalityTolerance' (lower bound on $\|\nabla D\|_\infty$), 'StepTolerance' (relative lower bound on the size of iteration steps) and 'ConstraintTolerance' (tolerances to which constraints are satisfied) are set to 10^{-12}. 500 randomly distributed points in the box $[-1,1]\times[-1,1]\times[-1,1]\times[-1,1]$ were used to initiate the minimization algorithm. Any points of local minimizers that violated the constraints or produced infeasible solutions were discarded. The remaining points were used to implement the Doyon and Dubé algorithm described in detail in Appendix A.

Leiva and Briozzo [20] converged 287 orbits for $-1.59407 < e < -1.58617$. Both the UPOs discovered in this work are newer for $e = -1.589070$, see Fig. 7. The values of $\overline{\Phi}$ calculated over previous UPOs at this energy level are 1.472 and 1.430 while that for the ones we discovered are 1.350 and 1.459. The minimum bound on $\overline{\Phi}$ corresponds to one of the UPOs we discovered, see Fig. 5. This strongly suggests that the UPO we converged is indeed extremal for the chosen $\Phi(\boldsymbol{a})$, and hence, most suited for a communication relay between Earth and the Moon at the prescribed energy level.

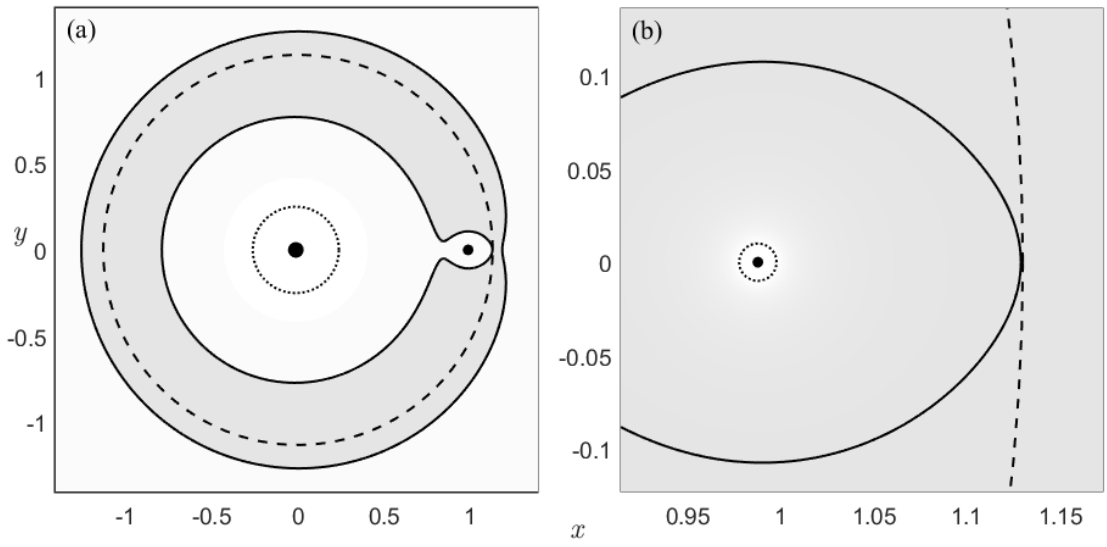


Fig. 4: Plots of the Hill's regions (areas in white), forbidden realms (area in grey) and Zero-velocity curves (—) in the x-y subspace for $e = -1.589070$. The region of interest bounded externally by the constraint G_1 (—) and internally by constraints G_2 and G_3 (- -) around Earth (●) and Moon (•). (b) Enlarged version of (a) around Moon (•)

.

7 Conclusions

Previous works on polynomial Sum-of-Squares optimization techniques to locate extremal periodic orbits for non-polynomial dynamical systems have relied on the following two approaches. While the first centres on the derivation of additional ODEs that represent the non-polynomial parts, the second approach requires the construction of a new system that is a polynomial approximation of the original ODE system. These approaches allow one to formulate the problem of finding optimal auxiliary functions using polynomial inequality and equality constraints. Specifically to application to PCR3BP, while both approaches do produce initial conditions that converged to a *candidate* UPO, known results [19, Section 7.4.4], [28, Section 7.3] have relied on *a priori* knowledge of a UPO in the desired topography to construct an "artificial" objective function. This limits the applicability of any converged UPO for space mission design.

As the first highlight of this paper, we introduce a novel technique of exploiting rational functions and approximations of radical parts together to obtain polynomial multiplier functions of the PCR3BP dynamical system. The problem statement is redefined with the distinguished feature of its applicability in the choice of objective functions that can be leveraged in the design of more practical space missions. We demonstrate the use of this technique in finding periodic orbits for the Earth-Moon system that minimizes the sum of squares of the distance of a satellite from the two primaries. We thus converged previously unknown UPOs for this system and showed that one of these UPOs is optimal in power

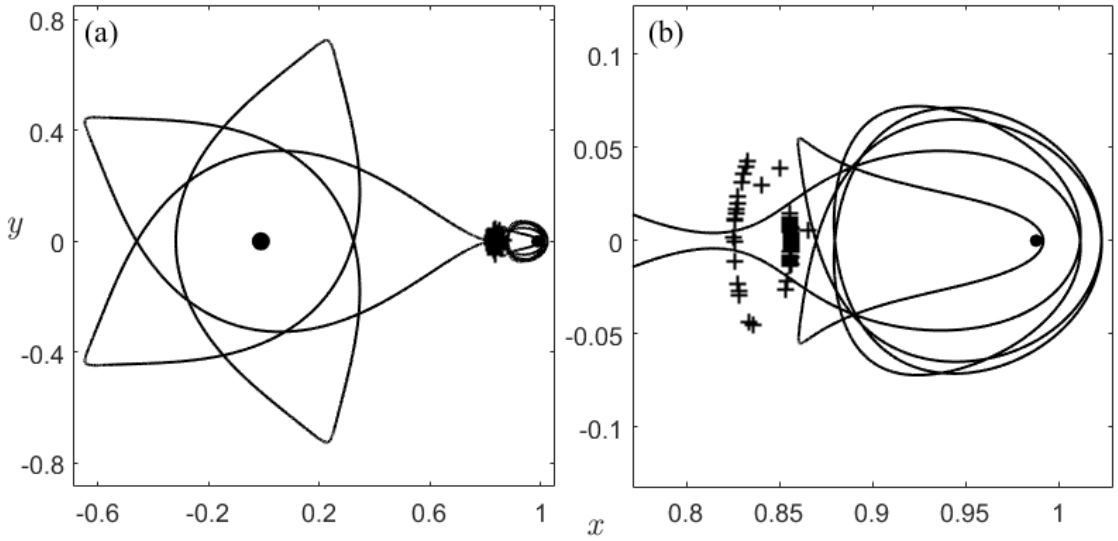


Fig. 5: (a) Earth (●), Moon (•), the converged extremal UPO−1 (—) with time period T = 22.44796989 for $\Phi(\boldsymbol{a})$ as defined in (28). Points of local minimizers (+) of $D(\boldsymbol{a})$ as defined in (21). (b) Enlarged version of (a) around the Moon.

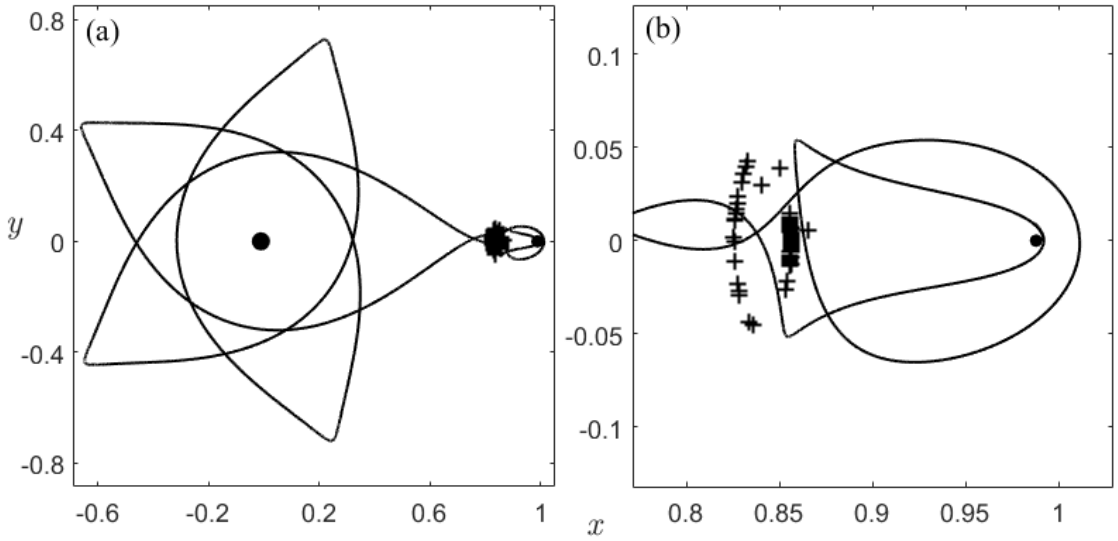


Fig. 6: (a) Earth (●), Moon (•), the converged extremal UPO−2 (—) with time period T = 17.96782688 for $\Phi(\boldsymbol{a})$ as defined in (28). Points of local minimizers (+) of $D(\boldsymbol{a})$ as defined in (21). (b) Enlarged version of (a) around the Moon.

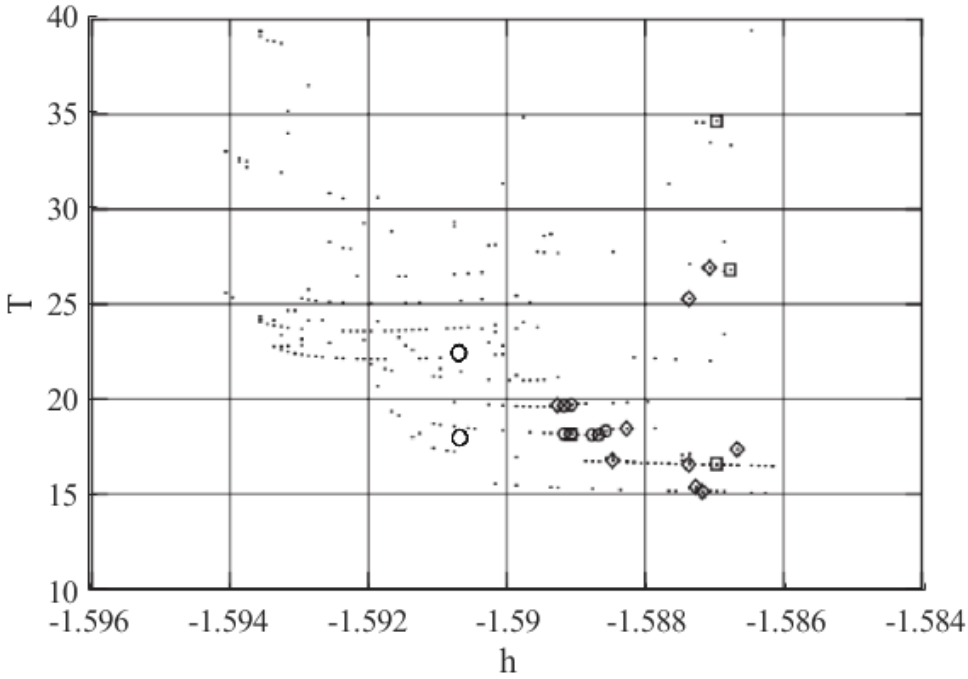


Fig. 7: Plot taken from Leiva and Briozzo [20, Fig. 2]. Period-energy (T–h) plot of the 287 periodic orbits denoted with squares (□), diamonds (◇) and dots (·) found in the numerical search. The circles (○) correspond to UPOs discovered in this work.

transmission for the communication relay problem. This is the second original contribution of this paper.

The third contribution is the demonstration of a close relationship between the convergence of a bound in YALMIP and the scaling factors employed in the problem formulation. YALMIP often runs into "numerical problems" while solving for sufficiently high degrees of Sum-of-Squares polynomials. While it is well-known that scaling ODE variables improves numerical conditioning in SOS optimizations, as noted by Goluskin [11], we found this technique to be particularly crucial in our setting. Here, we demonstrate using the Van der Pol oscillator that scaling a system down to different n-dimensional boxes with lengths less than unity can still ensure convergence for practically the highest desired degree of Sum-of-Squares polynomials employed. The necessity of employing this technique, particularly with application to systems like PCR3BP, is twofold. Firstly, a good rational approximation for non-polynomial parts of the ODEs renders one to work with high degrees of polynomials. Secondly, the convergence of unstable orbits requires a precise set of initial conditions. This is achieved with increasing degrees of auxiliary functions used to prove optimal bounds on the infinite-time average of an objective function.

The fourth and last contribution of this paper augments previous works that developed a numerical framework of direct nonlinear optimization to compute points inside a relevant sublevel set. These points provide good initial conditions for UPO computations. However, to uncover most of the target UPO, such minimization routines require starting with thousands of uniformly distributed random

initial conditions in the domain of interest. Each such point converges to desired sublevel sets and produces an approximate initial condition for UPO convergence. This requires a large computation time. Theoretical analysis suggests that each starting point travels through several other local minima but only terminates at the final point when other optimality tolerances employed in the minimization scheme are satisfied. These local minima can also be candidates for good initial conditions for UPO convergence. In this work, we show that this idea is viable through a newly developed numerical framework. All local minima lower than a threshold are captured even if the minimization routine continues to search for an optimal point that satisfies first-order optimality and step tolerances. This enables one to start with about a hundred randomly distributed points that eventually produce several thousand candidate initial conditions, thus uncovering the majority of the UPO.

Acknowledgements

- The author would like to sincerely thank Dr. Mayur V Lakshmi for valuable discussions on the application of the sum-of-squares SOS optimization technique. These interactions were highly educational and thought-provoking. The author also expresses gratitude to Dr. Giovanni Fantuzzi for insightful discussions regarding the applicability of the proposed method and for his time and effort in reviewing multiple versions of this work.
- The author gratefully acknowledges the support of the Commonwealth Scholarship Commission in the UK for sponsoring the MSc program at Imperial College London.

Appendix A Converging Extremal UPOs using Doyon and Dubé Algorithm

The methodology discussed so far in this chapter relies on initial conditions in state space lying close to the target UPO. Since UPOs for the Hamiltonian systems, such as PCR3BP, lie closely in a family of $(n-1)$-dimensional level set of Hamiltonian, it is necessary to develop algorithms that search for a target UPO in this reduced order state space. This section details one such algorithm of Doyon & Dubé [5]. The code is independently written by the author and is available to the reader upon request.

Let us revisit the set of n differential equations in state space given by (5). Let the flow of this system of equations at time t after integration from the initial condition $\boldsymbol{a}_0$ be $\boldsymbol{\varphi}(\boldsymbol{a}, t) \equiv \boldsymbol{a}(t)$. We are interested in developing a numerical framework for a mapping $\widetilde{\boldsymbol{a}}_k \mapsto \widetilde{\boldsymbol{a}}_{k+1} \equiv \mathcal{M}\left(\widetilde{\boldsymbol{a}}_k\right)$ at discrete times τ_k from this continuous flow where $\widetilde{\boldsymbol{a}}_k \in \mathbb{R}^m$ and $m < n$. Let the Poincaré Surface of Section (PSS), the hyperplane, be $\mathcal{P}$ with $n-1$ components be described by the following equations:

$$h(\boldsymbol{a}(t)) = \widetilde{\boldsymbol{\xi}}^T \left(\boldsymbol{a}(t) - \boldsymbol{a}_0\right), \tag{A1}$$

where $\boldsymbol{a}_0$ is a point on the hyperplane and $\widetilde{\boldsymbol{\xi}}$ is a vector orthogonal to this hyperplane. This hyperplane acts as one of the constraints for the discrete numerical mapping $\mathcal{M}\left(\widetilde{\boldsymbol{a}}_k\right)$. Specific to our aim of developing the algorithm for converging UPOs for the PCR3BP, we have a Hamiltonian constraint as well. For a total of n_c constraints between the different variables of the system, the numerical mapping will have $m = n - n_c$ components. For PCR3BP, we have $n = 4$, $n_c = 2$ and $m = 2$.

Furthermore, let the state of the system on the hyperplane at the k^{th} directed intersection be $\boldsymbol{a}_k$ and that in the numerically constructed discrete system $\mathcal{M}\left(\widetilde{\boldsymbol{a}}_k\right)$ be $\widetilde{\boldsymbol{a}}_k$ $\left(\boldsymbol{a}_k \in \mathbb{R}^n, (\widetilde{\boldsymbol{a}}_k, \mathcal{M}) \in \mathbb{R}^m\right)$. It is critical to note that the vector $\widetilde{\boldsymbol{a}}_k$ is simply a projection of $\boldsymbol{a_k}$ in a smaller subsystem formed by the removal of n_c components from the vector $\boldsymbol{a_k}$. Let us denote these n_c removed components by a vector $\breve{\boldsymbol{a}}_k \in \mathbb{R}^{n_c}$. This vector will depend on the choice of components of $\boldsymbol{a_k}$ that we keep in $\widetilde{\boldsymbol{a}}_k$ and we can, therefore, write this dependency as $\breve{\boldsymbol{a}}_k\left(\widetilde{\boldsymbol{a}}_k\right)$. By evaluating the state $\widetilde{\boldsymbol{a}}_k$, we can recover the n-dimensional state of the system on the hyperplane as

$$\boldsymbol{a_k} = \begin{pmatrix} \widetilde{\boldsymbol{a}}_k \\ \breve{\boldsymbol{a}}_k\left(\widetilde{\boldsymbol{a}}_k\right) \end{pmatrix} \tag{A2}$$

We can evaluate $\boldsymbol{a_k} \mapsto \boldsymbol{a}_{k+1}$ by integrating (5) from $\boldsymbol{a_k}$ for a particular time $t = \tau\left(\boldsymbol{a_k}\right) \equiv \tau_k$,

$$\boldsymbol{a}_{k+1} = \boldsymbol{\varphi}\left(\boldsymbol{a_k}, \tau\left(\boldsymbol{a_k}\right)\right) \tag{A3}$$

The challenge here is that we do not know *a priori* the time τ_k required to get back on the hyperplane. However, we do know that at this time, $\boldsymbol{a}_{k+1}$ is on the hyperplane and the flow meets a directed intersection condition. Strictly speaking, defining a directed intersection depends on the direction of the evolution of flow, but we can say an intersection is directed when the flow is going from negative to positive values of h without any loss of generality. Hénon's method [14] is a popular choice for similar exercises. We developed a numerical technique in conjunction with MATLAB's ODE suite to determine τ_k. The code that determines τ_k to machine accuracy is available to reader on request. We can summarize the numerical construction of discrete map $\mathcal{M}\left(\widetilde{\boldsymbol{a}}_k\right)$ to obtain $\widetilde{\boldsymbol{a}}_{k+1}$ from $\widetilde{\boldsymbol{a}}_k$ as follows:

1. First, to recover the n-dimensional state $\boldsymbol{a_k}$ we use the equations of constraint $\left(\breve{\boldsymbol{a}}\left(\widetilde{\boldsymbol{a}}_k\right)\right)$ in accordance with (A2).

2. Integrate the flow (5) from $\boldsymbol{a_k}$ using one of the above-mentioned numerical strategies up to $t = \tau_k$ when the flow is back on the defined hyperplane. The dynamical state at $t = \tau_k$ is $\boldsymbol{a}_{k+1}$.
3. Remove the $\breve{\boldsymbol{a}}_k$ components to obtain $\widetilde{\boldsymbol{a}}_{k+1}$ and repeat the procedure. The numerical map $\mathcal{M}\left(\widetilde{\boldsymbol{a}}_k\right)$ is thereby constructed from the flow which can be mathematically expressed as:

$$\mathcal{M}\left(\widetilde{\boldsymbol{a}}_k\right) = \boldsymbol{\varphi}\left(\boldsymbol{a_k}, \tau\left(\boldsymbol{a_k}\right)\right) = \boldsymbol{\varphi}\left(\widetilde{\boldsymbol{a}}_k, \breve{\boldsymbol{a}}_k\left(\widetilde{\boldsymbol{a}}_k\right), \tau\left(\widetilde{\boldsymbol{a}}_k, \breve{\boldsymbol{a}}_k\left(\widetilde{\boldsymbol{a}}_k\right)\right)\right) \tag{A4}$$

It is to be re-emphasized that the points of minimization of δ-suboptimal D lying in the set $\mathcal{S}\varepsilon$, as defined in (16), are only approximations of initial conditions for a UPO. In such cases, it is also important to determine the Jacobian matrix $\mathbf{J}\left(\widetilde{\boldsymbol{a}}_k\right)$ of the numerical map $\mathcal{M}\left(\widetilde{\boldsymbol{a}}_k\right)$.

$$\delta\widetilde{\boldsymbol{a}}_{k+1} = \mathbf{J}\left(\widetilde{\boldsymbol{a}}_k\right)\delta\widetilde{\boldsymbol{a}}_k \tag{A5}$$

$\mathbf{J}\left(\widetilde{\boldsymbol{a}}_k\right)$ can be obtained by differentiating the map $\mathcal{M}(\widetilde{\boldsymbol{a}}_k)$ with respect to $\widetilde{\boldsymbol{a}}_k$. We can see from (A4) the dependence of the $\mathcal{M}(\widetilde{\boldsymbol{a}}_k)$ on the constrained variables $\left(\breve{\boldsymbol{a}}\left(\widetilde{\boldsymbol{a}}_k\right)\right)$ which in turn are dependent on $\widetilde{\boldsymbol{a}}_k$. For this reason, we must take a total derivative. From (A4), using the chain rule for differentiation, we get,

$$\begin{aligned}
\mathbf{J}\left(\widetilde{\boldsymbol{a}}_k\right) &= \frac{\mathrm{d}}{\mathrm{d}\widetilde{\boldsymbol{a}}}\mathcal{M}\left(\widetilde{\boldsymbol{a}}_k\right)\Big|_{\widetilde{\boldsymbol{a}}_k} = \frac{\mathrm{d}}{\mathrm{d}\widetilde{\boldsymbol{a}}}\boldsymbol{\varphi}\left(\widetilde{\boldsymbol{a}}_k, \breve{\boldsymbol{a}}_k\left(\widetilde{\boldsymbol{a}}_k\right), \tau\left(\widetilde{\boldsymbol{a}}_k, \breve{\boldsymbol{a}}_k\left(\widetilde{\boldsymbol{a}}_k\right)\right)\right)\Big|_{\boldsymbol{a}_k,\tau_k} \\
&= \mathrm{D}_{\widetilde{\boldsymbol{a}}}\boldsymbol{\varphi}|_{\boldsymbol{a}_k,\tau_k} + \mathrm{D}_{\breve{\boldsymbol{a}}}\boldsymbol{\varphi}|_{\boldsymbol{a}_k,\tau_k}\mathrm{D}_{\widetilde{\boldsymbol{a}}}\breve{\boldsymbol{a}}\Big|_{\boldsymbol{a}_k} + \left(\frac{\partial\boldsymbol{\varphi}}{\partial\tau}\frac{\mathrm{d}\tau}{\mathrm{d}\widetilde{\boldsymbol{a}}}\right)\Big|_{\boldsymbol{a}_k,\tau_k} \\
&= \mathrm{D}_{\widetilde{\boldsymbol{a}}}\boldsymbol{\varphi}|_{\boldsymbol{a}_k,\tau_k} + \left(\boldsymbol{f}(\boldsymbol{\varphi})\mathrm{D}_{\widetilde{\boldsymbol{a}}}\tau\right)\big|_{\boldsymbol{a}_k,\tau_k} + \left(\left(\mathrm{D}_{\breve{\boldsymbol{a}}}\boldsymbol{\varphi} + \boldsymbol{f}(\boldsymbol{\varphi})\mathrm{D}_{\breve{\boldsymbol{a}}}\tau\right)\mathrm{D}_{\widetilde{\boldsymbol{a}}}\breve{\boldsymbol{a}}\right)\Big|_{\boldsymbol{a}_k,\tau_k}
\end{aligned} \tag{A6}$$

where D denotes the partial derivatives. We can evaluate the first term by using the variational equations associated with the system of equations (5) given by:

$$\frac{\mathrm{d}\mathbf{J}_{\bar{\varphi}}(\boldsymbol{a}, t)}{\mathrm{d}t} = \mathrm{D}_{\boldsymbol{a}}\boldsymbol{f}(\boldsymbol{a})\mathbf{J}_{\varphi}(\boldsymbol{a}, t) \tag{A7}$$

The solution of integration of (A7) starting from the initial condition $\mathbf{J}_{\varphi}(\boldsymbol{a_0}) = \mathbf{1}$ to $t = \tau_k$ will generate the $n \times n$ Jacobian matrix $\mathbf{J}_{\varphi}(\boldsymbol{a}_k, \tau_k)$. It quantifies how much perturbation evolves with time in the neighbourhood of $\boldsymbol{\varphi}(\boldsymbol{a}_k, \tau_k)$ and takes the explicit form:

$$\mathbf{J}_{\varphi}(\boldsymbol{a}_k, \tau_k) \equiv \mathrm{D}_{\boldsymbol{a}}\boldsymbol{\varphi}(\boldsymbol{a}_k, \tau_k) = \begin{pmatrix} \frac{\partial\varphi^1}{\partial a_k^1} & \cdots & \frac{\partial\varphi^1}{\partial a_k^n} \\ \vdots & \ddots & \vdots \\ \frac{\partial\varphi^n}{\partial a_k^1} & \cdots & \frac{\partial\varphi^n}{\partial a_k^n} \end{pmatrix} \tag{A8}$$

$D_{\widetilde{\boldsymbol{a}}}\boldsymbol{\varphi}|_{\boldsymbol{a}_k,\tau_k}$ is then the top-left $m \times m$ submatrix of $\mathbf{J}_{\varphi}\left(\boldsymbol{a_k}, \tau_k\right)$. The second term $\boldsymbol{f}(\boldsymbol{\varphi})$ is computed by taking the first m components of the vector obtained by evaluating the right-hand side of (5) at $\boldsymbol{\varphi}\left(\boldsymbol{a}_k, \tau_k\right) = \boldsymbol{a}_{k+1}$. For evaluating the rest of the terms, note that the state $\delta\boldsymbol{a}_{i+1}$ must lie on the Hyperplane. This requires $\boldsymbol{\xi}^{\top}\delta\boldsymbol{a}_{i+1} = 0$. Accurate to first-order, we can expand this relation to obtain:

$$\boldsymbol{\xi}^{\top}\left(\frac{\partial\boldsymbol{\varphi}}{\partial\boldsymbol{a}} + \frac{\partial\boldsymbol{\varphi}}{\partial\tau}\frac{\mathrm{d}\tau}{\mathrm{d}\boldsymbol{a}}\right)\Big|_{\boldsymbol{a}_k,\tau_k} = \mathbf{0} \tag{A9}$$

From $\partial\varphi/\partial\tau = \boldsymbol{f}(\boldsymbol{\varphi})$, we have,

$$\left.\frac{\partial\tau}{\partial\boldsymbol{a}}\right|_{\boldsymbol{a}_k,\tau_k} = \frac{-\left.\boldsymbol{\xi}^\top \frac{\partial\boldsymbol{\varphi}}{\partial\boldsymbol{a}}\right|_{\boldsymbol{a}_k,\tau_k}}{\left.\boldsymbol{\xi}^\top \boldsymbol{f}(\boldsymbol{\varphi})\right|_{\boldsymbol{a}_k,\tau_k}} \tag{A10}$$

$\mathrm{D}_{\breve{\boldsymbol{a}}}\tau$ is the line vector composed of the last n_c elements of $\mathrm{D}_{\boldsymbol{a}}\tau|_{\boldsymbol{a}_k,\tau_k}$ (A10). The last term of (A6) is important in the case of Hamiltonian systems, as we will later see with application to PRC3BP, where the removed components $\breve{\boldsymbol{a}}$ depends on $\tilde{\boldsymbol{a}}$. $\mathrm{D}_{\breve{a}}\boldsymbol{\varphi}$ is the $m \times n_c$ submatrix of the variational matrix $\mathbf{J}_\varphi(\boldsymbol{a}_k, \tau_k)$. And finally, $\mathrm{D}_{\widetilde{\boldsymbol{a}}}\breve{\boldsymbol{a}}$ is an $n_c \times D$ matrix consisting of derivatives of $\breve{\boldsymbol{a}}$ with respect to $\widetilde{\boldsymbol{a}}$ as given by the equations of constraints.

Our goal, now, is to use a simple form of the Newton-Raphson algorithm to find the root of:

$$\boldsymbol{A}(\widetilde{\boldsymbol{a}}) - \widetilde{\boldsymbol{a}} = \mathbf{0} \tag{A11}$$

It is useful to note here that equation A11 onsists of the initial m components of the full n-dimensional state vector $\boldsymbol{a}$. Our aim is to perform Newton-Raphson iterations in the following manner [1, Section 4.6]. At each step k, we update $\widetilde{\boldsymbol{a}}$ to $\widetilde{\boldsymbol{a}}_{k+1}$ by adding the correction $\delta\widetilde{\boldsymbol{a}}_k$:

$$\widetilde{\boldsymbol{a}}_{k+1} = \widetilde{\boldsymbol{a}}_k + \delta\widetilde{\boldsymbol{a}}_k \tag{A12}$$

To calculate $\delta\widetilde{\boldsymbol{a}}_k$, we can solve a linear system composed of m equations:

$$\left[I - \boldsymbol{J}\left(\widetilde{\boldsymbol{a}}_k\right)\right]\delta\boldsymbol{a}_k = \boldsymbol{A}\left(\widetilde{\boldsymbol{a}}_k\right) - \widetilde{\boldsymbol{a}}_k \tag{A13}$$

Importantly, it is worth noting once again that the mapping $\boldsymbol{A}(\widetilde{\boldsymbol{a}})$ actually depends implicitly on the complete n-dimensional state $\boldsymbol{a}$. Hence, we can easily recover the $n_c = n - m$ missing components using the constraint equations $\breve{\boldsymbol{a}}(\widetilde{\boldsymbol{a}})$.

References

[1] Uri M Ascher, Robert MM Mattheij, and Robert D Russell. *Numerical solution of boundary value problems for ordinary differential equations*. SIAM, 1995.

[2] Paula Betriu, Manel Soria, Jordi L Gutiérrez, Marcel Llopis, and Antoni Barlabé. An assessment of different relay network topologies to improve earth–mars communications. *Acta Astronautica*, 206:72–88, 2023.

[3] Grigoriy Blekherman, Pablo A Parrilo, and Rekha R Thomas. *Semidefinite optimization and convex algebraic geometry*. SIAM, 2012.

[4] Sergei I Chernyshenko, P Goulart, D Huang, and Antonis Papachristodoulou. Polynomial sum of squares in fluid dynamics: a review with a look ahead. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 372(2020):20130350, 2014.

[5] B Doyon and LJ Dubé. On jacobian matrices for flows. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 15(1), 2005.

[6] J-P Eckmann and David Ruelle. Ergodic theory of chaos and strange attractors. *Reviews of modern physics*, 57(3):617, 1985.

[7] Giovanni Fantuzzi, David Goluskin, Deqing Huang, and Sergei I Chernyshenko. Bounds for deterministic and stochastic dynamical systems using sum-of-squares optimization. *SIAM Journal on Applied Dynamical Systems*, 15(4):1962–1988, 2016.

[8] Robert W Farquhar. Halo-orbit and lunar-swingby missions of the 1990's. *Acta Astronautica*, 24:227–234, 1991.

[9] David C Folta, Mark Woodard, Kathleen Howell, Chris Patterson, and Wayne Schlei. Applications of multi-body dynamical environments: The artemis transfer trajectory design. *Acta Astronautica*, 73:237–249, 2012.

[10] David Goluskin. Bounding averages rigorously using semidefinite programming: mean moments of the lorenz system. *Journal of Nonlinear Science*, 28:621–651, 2018.

[11] David Goluskin. Bounding extrema over global attractors using polynomial optimisation. *Nonlinearity*, 33(9):4878, 2020.

[12] David Goluskin and Giovanni Fantuzzi. Bounds on mean energy in the kuramoto–sivashinsky equation computed using semidefinite programming. *Nonlinearity*, 32(5):1705, 2019.

[13] Gerard Gómez, Wang S Koon, Martin W Lo, Jerrold E Marsden, Josep Masdemont, and Shane D Ross. Connecting orbits and invariant manifolds in the spatial restricted three-body problem. *Nonlinearity*, 17(5):1571, 2004.

[14] Michel Hénon. On the numerical computation of poincaré maps. *Physica D: Nonlinear Phenomena*, 5(2-3):412–414, 1982.

[15] Kathleen C Howell and Henry J Pernicka. Numerical determination of lissajous trajectories in the restricted three-body problem. *Celestial mechanics*, 41(1-4):107–124, 1987.

[16] Wang Sang Koon, Martin W Lo, Jerrold E Marsden, and Shane D Ross. Dynamical systems, the three-body problem and space mission design. In *Equadiff 99: (In 2 Volumes)*, pages 1167–1181. World Scientific, 2000.

[17] Mayur V Lakshmi, Giovanni Fantuzzi, Sergei I Chernyshenko, and Davide Lasagna. Finding unstable periodic orbits: A hybrid approach with polynomial optimization. *Physica D: Nonlinear Phenomena*, 427:133009, 2021.

[18] Mayur V Lakshmi, Giovanni Fantuzzi, Jesús D Fernández-Caballero, Yongyun Hwang, and Sergei I Chernyshenko. Finding extremal periodic orbits with polynomial optimization, with application to a nine-mode model of shear flow. *SIAM Journal on Applied Dynamical Systems*, 19(2):763–787, 2020.

[19] Mayur Venkatram Lakshmi. *Finding Unstable Periodic Orbits for Nonlinear Dynamical Systems Using Polynomial Optimisation*. PhD thesis, Imperial College London, 2022.
[20] AM Leiva and CB Briozzo. Control of chaos and fast periodic transfer orbits in the earth–moon cr3bp. *Acta Astronautica*, 58(8):379–386, 2006.
[21] Johan Lofberg. Pre-and post-processing sum-of-squares programs in practice. *IEEE transactions on automatic control*, 54(5):1007–1011, 2009.
[22] Jeff Moehlis, Holger Faisst, and Bruno Eckhardt. A low-dimensional model for turbulent shear flows. *New Journal of Physics*, 6(1):56, 2004.
[23] Katta G Murty and Santosh N Kabadi. Some np-complete problems in quadratic and nonlinear programming. Technical report, 1985.
[24] Edward Ott, Celso Grebogi, and James A Yorke. Controlling chaos. *Physical review letters*, 64(11):1196, 1990.
[25] Pablo A Parrilo. *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization*. California Institute of Technology, 2000.
[26] Pablo A Parrilo. Semidefinite programming relaxations for semialgebraic problems. *Mathematical programming*, 96:293–320, 2003.
[27] Bruce Reznick. Some concrete aspects of hilbert's 17th problem. *Contemporary mathematics*, 253:251–272, 2000.
[28] Vinay Sharma. Locating extremal periodic orbits in astrodynamical systems using polynomial sum-of-squares optimization. 2021, Birla Institute of Technology and Science, Pilani.
[29] Victor Szebehely and E Grebenikov. Theory of orbits-the restricted problem of three bodies. *Soviet Astronomy, Vol. 13, p. 364*, 13:364, 1969.
[30] Jim Taylor, editor. *Deep Space Communications*. John Wiley & Sons, Ltd, 2016.
[31] Ian Tobasco, David Goluskin, and Charles R Doering. Optimal bounds and extremal trajectories for time averages in nonlinear dynamical systems. *Physics Letters A*, 382(6):382–386, 2018.