A. Capannolo, A. Pasquale, M. Lavagna
High-Order Polynomial Continuation Method for Trajectory Design in Non-Keplerian Environments
Celestial Mechanics and Dynamical Astronomy, Vol. 133, N. 10, 48, 2021, p. 1-21
doi:10.1007/s10569-021-10046-4

High-Order Polynomial Continuation Method for Trajectory Design in Non-Keplerian Environments

A. Capannolo* · A. Pasquale* · M. Lavagna*

the date of receipt and acceptance should be inserted later

Abstract Orbit generation in Non-Keplerian environments poses some challenges related to the complex dynamical nature in which such trajectory exist. The absence of a parametric representation of the orbits requires an iterative approach to define families. Simple methods exist to fulfill such task, however, they are based on local information and prone to convergence/speed problems. A polynomial-based scheme is proposed to improve the search of the solutions along the orbital families, enhancing the overall speed of the process, while avoiding convergence issues. The scheme is tested in the framework of Earth-Moon system, and performances are discussed and compared to classical approaches.

1 Introduction

The design of orbits in Non-Keplerian environments is characterized by intrinsic complexity of mathematical and numerical nature. The absence of a parametrized representation of such orbits require the mission designer to adopt iterative procedures, leveraging a prediction-correction scheme, which introduce computational burden, translating into relatively long-time simulations. In particular, the problem **requires identification of** a starting condition, and gradually move across the state space until the final desired orbit is achieved [1].

Several well-known methods are employed to achieve such goal, which typically leverage local information of the orbital families being generated. Among the others, the most used methods are the *natural parameter* continuation, where a single physical parameter (e.g. a state component or the orbital period) is perturbed to generate a new guess, and the first-order methods, such *pseudo-arclength*, which leverages local tangents to the family [4,9,3,5]. The main drawback related to the aforementioned approaches is the high sensitivity

*Politecnico di Milano, Aerospace Science ad Technologies Department

to the state variation, therefore small steps are needed to ensure convergence at each iteration. The length of the step, in turn, affects the speed of the process.

In the present paper, a higher order continuation scheme based on polynomials is explored [11,6,10]. The Circular Restricted Three-Body Problem (CR3BP), is used as dynamical framework to assess improvements in the families generation scheme, with respect to classical approaches. The scheme leverages the more accurate guess coming from the polynomial, to increase and adapt the continuation step length, and ultimately ensure a faster generation of the orbital family. Moreover, the presented approach can be efficiently embedded to a bifurcation detection scheme, commonly used in the context of the CR3BP to identify new orbital families [16,2].

The paper is structured as follows. **Section 2 introduces** the dynamical environment and the classical prediction-correction approach. Section 3 contains a full description of the proposed algorithm, with detailed discussion on each component. In Section 4 results of the approach are shown and critically discusses for some orbital families in the Earth-Moon binary system. Finally, Section 5 draws conclusions about the approach, highlighting strengths and weaknesses, and proposing potential improvements for the scheme.

2 Background & Tools

2.1 The Circular Restricted Three-Body Problem

The classical Circular Restricted Three-Body Problem (CR3BP) consider the motion of three bodies as point masses and studies the motion of the third body (m), that moves under the gravitational attraction of the primaries $\mathscr{P}_1$ and $\mathscr{P}_2$ that are constrained to move on a circular orbit around the Center of Mass (G) of their two-body system. It is common to express the equation of motion for the CR3BP in the so called *synodic reference frame* [14]. In this frame, the position of the primaries is fixed in time on the x (synodic) axis. Thus, the frame rotates with angular velocity equal to that of the primaries (two-body motion). Eq. 1 shows the equations of motion in the non dimensional form:

$$\begin{cases} \ddot{x} - 2\dot{y} = \mathcal{U}_x \\ \ddot{y} + 2\dot{x} = \mathcal{U}_y \\ \ddot{z} = \mathcal{U}_z \end{cases} \tag{1}$$

Here $\dot{(\cdot)}$ and $\ddot{(\cdot)}$ denotes the first and the second derivatives with respect to the (non-dimensional) time, while $\mathcal{U}_{(\cdot)}$ denotes the partial derivative of the pseudopotential function $\mathcal{U}$ with respect to the variable $(\cdot)$. Here the pseudopotential is defined as:

$$\mathcal{U}^{(\mathrm{cr3bp})} = \frac{1}{2}(x^2 + y^2) + \frac{1-\mu}{r_1} + \frac{\mu}{r_2} \tag{2}$$

where r_1 and r_2 represent the distance of the particle from the primaries and μ is the mass ratio between the **secondary attractor** and the whole system, which is defined as:

$$\mu = \frac{m_2}{m_1 + m_2} \tag{3}$$

2.2 Predictor-Corrector Continuation Schemes

Curves implicitly defined by underdetermined **systems** of equations are considered in the context of numerical continuation methods. In particular, consider a smooth map $\mathcal{G} : \mathbb{R}^{n+1} \to \mathbb{R}^n$, such that:

$$\mathcal{G}(\boldsymbol{\chi}) = 0 \tag{4}$$

then continuation algorithms have the objective to find the solution path P of Eq. 4 such that:

$$P = \{\boldsymbol{\chi} : \boldsymbol{\chi} = \psi(t), t \in I\} \tag{5}$$

where I is an interval of interest for the continuation parameter t, where the path is not rank-deficient [10,1]. Continuation algorithms then generate a set of points $\{(t_k, \boldsymbol{\chi}_k)\}_{k=0}^{N}$ where $\boldsymbol{\chi}_0$ is a known solution of Eq. 4 and $\boldsymbol{\chi}_k$ are on the path P.

A class of largely used path following algorithms are the so called *predictor-corrector* schemes. The use of this technique, which is largely applied to the solution of Ordinary Differential Equations (ODE) arise from the fact that Eq. 4 can be rewritten in a more convenient **form** as [10]:

$$\frac{d\boldsymbol{\chi}}{dt} = T(\boldsymbol{\chi}), \quad \boldsymbol{\chi}(t_0) = \boldsymbol{\chi}_0 \tag{6}$$

where $T(\boldsymbol{\chi})$ is a properly oriented non-zero null vector of $D_\chi \mathcal{G}(\boldsymbol{\chi})$. In general, those methods consist of the two steps:

1. a *predictor step*: where an approximate step along the curve is performed;
2. a *corrector step*: where are performed one or more iterative steps for solving the constraint equation, $\mathbf{F}(\boldsymbol{\chi}) = 0$, which brings the predicted point back to the curve.

In this work the correction step is performed with a shooting scheme, which is described in Sec. 2.3, while the prediction is performed with different schemes, including:

- a **natural parameter** scheme, in which the continuation parameter t is assumed to be a *physical* quantity (e.g. position, velocity, mass-ratio, ...).
- a **first-order** scheme, where the tangent to the path P is computed using a first-order, forward Euler method. This method requires to have at least *two* known solutions along the continuation path.
- a **pseudo arc-length** scheme, where the tangent to the path is exploited by the null space of the Jacobian of the map $\mathcal{G}$.

2.3 Differential Correction: The Single Shooting Method

The generation of periodic or quasi-periodic orbits in the context of the CR3BP relies on the use of methods for the solution of Boundary-Value Problems. Among the others, orthogonal collocation methods and shooting methods are the most used for this purpose. The *Single Shooting Method* is commonly used due to its simplicity [7]. In general, consider a set of free-variables, $\boldsymbol{\chi} \in \mathbb{R}^d$:

$$\boldsymbol{\chi} = \begin{bmatrix} \chi_1 \\ \vdots \\ \chi_d \end{bmatrix} \tag{7}$$

To ensure that the designed trajectory possess some desired characteristics, the free variables vector is subject to c scalar constraint equations satisfying $\boldsymbol{F}(\boldsymbol{\chi}) = 0$:

$$\boldsymbol{F}(\boldsymbol{\chi}) = \begin{bmatrix} F_1(\boldsymbol{\chi}) \\ \vdots \\ F_c(\boldsymbol{\chi}) \end{bmatrix} = 0 \tag{8}$$

The goal of this approach it the numeric computation of a solution $\boldsymbol{\chi}^*$ that satisfies the constrained equation $\boldsymbol{F}(\boldsymbol{\chi}^*) = 0$ within some numerical accuracy ε_{corr}.

Such objective is achieved through an iterative procedure, by locally linearizing Eq. 8 with respect to the problem's variables, and computing the new variables set as:

$$\boldsymbol{\chi}_{k+1} = \boldsymbol{\chi}_k - \left[\frac{\partial \boldsymbol{F}(\boldsymbol{\chi})}{\partial \boldsymbol{\chi}}\right]_k^{-1} \boldsymbol{F}(\boldsymbol{\chi}_k) \tag{9}$$

The process is continued until the set of constraints falls below the specified accuracy:

$$||\boldsymbol{F}(\boldsymbol{\chi}_{k+1})|| < \varepsilon_{corr} \tag{10}$$

3 Algorithms description

The proposed scheme is based on the typical "Prediction-Correction" approach; however, its novelty lies within the blocks of such loop. In particular, the structure of the algorithm is designed to increase the speed in the generation of full orbital families. Such goal is attained through two aspects of the scheme:

- Improving the goodness of the prediction, to ensure a fast convergence of the guess correction sub-problem.
- Allowing larger continuation steps, to rapidly develop the orbital family.

Such targets are strongly related to each other, as a larger step typically implies a loss in the accuracy of the guess, hindering the capability of the corrector of converging to a feasible solution (within the time or number of iterations given). The purpose of the revised continuation scheme is to exploit the information of the previously corrected solutions to aid the prediction of the new orbit to be generated, and allow to increase the step to gain speed in the process. The overall scheme's structure is depicted in Fig. 1, and single blocks are hereafter described in detail.

3.1 Initialization

The initialization of the process requires the knowledge of at least two orbits. **The first orbit is directly defined from the designer's choice (e.g. an equilibrium point, a Keplerian guess, ecc.), while the second solution can be obtained through a "classical" continuation-correction scheme, leveraging the first orbit.** The adopted correction scheme leverages the Levenberg-Marquardt algorithm (described in 3.2.2). The two computed solutions are collected together to form the initial states stack:

$$\mathbb{X}_{,2} = \begin{Bmatrix} \boldsymbol{\chi}_{,1}^T \\ \boldsymbol{\chi}_{,2}^T \end{Bmatrix} \tag{11}$$

where the single vector $\boldsymbol{\chi}$ comprises the 6 elements initial state of the orbit ($\boldsymbol{X}$), and its period (τ):

$$\boldsymbol{\chi} = \begin{bmatrix} \boldsymbol{X} \\ \tau \end{bmatrix} \tag{12}$$

A further parameter required for the proposed scheme is the curvilinear abscissa (the variable of the prediction polynomial, described in 3.2.1). The curvilinear abscissa can have arbitrary initial values, as the computation of polynomial coefficients will adapt to it, therefore the abscissa stack reads:

$$\mathbb{T}_{,2} = \begin{Bmatrix} 0 \\ \Delta t \end{Bmatrix} \tag{13}$$

with Δt being the arbitrary value, acting as initial step for the continuation loop.

Finally, the initial polynomial degree is determined by the amount of initial computed orbits. In the minimal case scenario (two initial orbits), a first degree polynomial is required ($n = 1$).

3.2 Continuation loop

3.2.1 Guess generation

Guess are generated starting from the states and abscissa stacks, leveraging an approximating polynomial function [13]. Depending on the current degree

of the polynomial "n", the last $n+1$ states, and corresponding abscissas, are extracted from the stack. First, the abscissa subset is pivoted to have the last corrected orbit at the zero value (the previous n orbits will have negative abscissa values). This approach allows to always have the next prediction at low abscissa values (positive and near zero), and avoid possible numerical issues when the polynomial has a high degree. Then, Vandermonde matrix is built to collect all abscissa steps, elevated to all powers from zero up to the polynomial's degree. At each **iteration** "k", the matrix reads:

$$A_{ri,k} = (\mathbb{T}_{r,k} - \mathbb{T}_{N,k})^{n-i+1}, \quad r = (N-n) : N,\, i = 1 : n+1; \tag{14}$$

with N being the full length of the abscissa stack. The polynomial coefficients vector is then computed by inverting the linear system relating the states to the abscissa powers:

$$\boldsymbol{C}_{ij,k} = A^{-1}_{ri,k}\mathbb{X}_{rj,k}, \quad r = N-n : N,\, i = 1 : n+1,\, j = 1 : 7 \tag{15}$$

The new state guess is finally obtained by constructing the polynomial with the coefficients from Eq.15 and the current abscissa step Δt:

$$\boldsymbol{\chi}_{j,k+1}|_{guess} = \Delta t^{n-i+1}\boldsymbol{C}_{ij,k}, \quad i = 1 : n+1,\, j = 1 : 7 \tag{16}$$

3.2.2 Correction

The correction of the guess follows the classical scheme presented in 2.3. A set of conditions to be nullified ($\boldsymbol{F}(\boldsymbol{\chi}) = 0$) is defined:

$$\boldsymbol{F}_{,k,l} = \begin{bmatrix} \boldsymbol{\varphi}(\boldsymbol{X}_{,k,l}, \tau_{,k,l}) - \boldsymbol{X}_{,k,l} \\ \boldsymbol{\sigma}(\boldsymbol{\chi}_{,k,l}) \end{bmatrix} \tag{17}$$

with the subscript l indicating the iterations of the inner correction loop, $\boldsymbol{\varphi}$ being the propagated initial state for one orbital period and $\boldsymbol{\sigma}$ representing the set of *Poincaré phase conditions*. The first row of Eq.17 is the connection between initial and final (propagated) state in one orbital period; such condition ensures the periodicity of the trajectory. The phase conditions are a set of additional constraints to fix part of the states and ensure the uniqueness of the solution. Such conditions are tailored on the specific orbit family, therefore they are explicitly discussed in Section 4.1.

Following the Newton scheme, **the linearization** from Eq.9 is iteratively inverted to extract the new state:

$$\boldsymbol{\chi}_{,k+1,l+1} = \boldsymbol{\chi}_{,k+1,l} - \left(\mathbb{J}^T\mathbb{J}\right)^{-1}\mathbb{J}^T\boldsymbol{F}(\boldsymbol{\chi}_{,k+1,l}) \tag{18}$$

where $\mathbb{J}$ is the Jacobian matrix, defined as the set of partial derivatives of the conditions $\boldsymbol{F}(\boldsymbol{\chi}_{,k+1,l})$ with respect to the problem variables $\boldsymbol{\chi}_{,k+1,l}$. To gain stability in the correction process, the Newton method is modified into the Levenberg-Marquardt scheme:

$$\boldsymbol{\chi}_{,k+1,l+1} = \boldsymbol{\chi}_{,k+1,l} - \left(\mathbb{J}^T\mathbb{J} + \lambda\mathbb{I}\right)^{-1}\mathbb{J}^T\boldsymbol{F}(\boldsymbol{\chi}_{,k+1,l}) \tag{19}$$

where λ represents a damping parameter which is updated at each iteration: if the new residual $\|\boldsymbol{F}(\boldsymbol{\chi}_{,k+1,l+1})\|_\infty$ is smaller than the previous one, λ is reduced by an order of magnitude, otherwise it is increased by the same quantity. As λ approaches 0, the equation becomes closer to the Newton's formula; vice versa, with λ increasing, the process approaches the gradient method.

3.2.3 Step/Degree update

Once the corrected state is obtained, checks on the goodness of the iteration are performed. In particular, the iteration is accepted only if both the correction is successful (the residual is below a specified tolerance), and the prediction (guess) is close enough to the final corrected value (the maximum "distance" between predicted and corrected elements is below a specified tolerance). This is evaluated as:

$$\begin{aligned} e_{,k+1}|_c &= \|\boldsymbol{F}(\boldsymbol{\chi}_{,k+1})\|_\infty < tol|_c \\ e_{,k+1}|_p &= \|\boldsymbol{\chi}_{,k+1} - \boldsymbol{\chi}_{,k+1}|_{guess}\|_\infty < tol|_p \end{aligned} \tag{20}$$

In case one or both conditions are not satisfied, the iteration is rejected, and the continuation step is modified to a lower and more conservative value. In particular, since the error is higher than the tolerance, their ratio is leveraged to reduce the step:

$$\begin{aligned} r &= \left(\frac{tol}{e_{,k+1}}\right)\Big|_p \\ \Delta t|_{new} &= r\Delta t|_{old} \end{aligned} \tag{21}$$

To avoid excessive reduction due to a very bad prediction, the ratio r is always bounded within the range $[0.5, 0.9]$.

If both conditions of Eq.20 are satisfied, the iteration is accepted, and the algorithm proceeds to the update of the polynomial degree and of the abscissa step. The scheme for step and degree update is inspired by the Adams-Bashforth-Moulton predictor-corrector algorithm for the propagation of ordinary differential equations [12]. The scheme follows a sequence of three cascade-connected steps:

- Polynomial degree reduction
- Step increment
- Polynomial degree increment

Polynomial degree reduction First, the algorithm checks if reducing the polynomial degree leads to a better state prediction (particularly true for plateau or slowly-varying regions of the orbital family). This is achieved through an inner `while` loop. A new prediction error "$e^-_{,k+1}|_p$" is computed as in Eq.20, where a $\boldsymbol{\chi}_{,k+1|_{guess}}$ is updated through Eq.16 with n reduced by one. If $e^-_{,k+1}|_p \leq e_{,k+1}|_p$, the new error and degree are accepted and the process is repeated, otherwise the latest accepted degree reduction holds.

Notice that this procedure allows to find a local minimum for the prediction error, while lower degrees may still produce better results. Nevertheless, the lack of a full search of the minimum across all possible polynomial degrees is compensated by a reduced amount of iterations, which can greatly speed up the outer continuation loop of the scheme.

Step increment A possible increment of the continuation step is evaluate only after the degree reduction is performed. In this way, the final prediction error is ensured to be the lowest (locally), providing better conditions for increasing the step (and, accordingly, the continuation speed). In particular, the step is doubled if the latest prediction error is lower than 50% of the set prediction tolerance, otherwise it remains unchanged.

Polynomial degree increment As last step, it is evaluated if increasing the degree of the polynomial may improve the prediction accuracy. The reason behind treating the degree increment separately from the degree reduction and, more important, after the step update, is to maximize the numerical stability of the continuation process. In fact, high-degree polynomials are more sensitive to the abscissa step, therefore two assessments are required before considering increasing the degree:

- **Better predictions exist with lower degree**. If both higher and lower degree polynomials provide a better prediction, it is always preferred to reduce the degree (because of the lesser sensitivity to the step)
- **Step is increased**. If prediction error is low enough to increase the step, it is not convenient to increase polynomial degree as well, as a combined effect of longer step and higher degree may cause large errors in the subsequent iteration.

Furthermore, fast degree increment is prevented by allowing a maximum increment of one for each continuation iteration. Hence, if degree is not reduced, the step is unchanged, and $e^{+}_{,k+1}|_p \leq e_{,k+1}|_p$ (where $e^{+}_{,k+1}|_p$ is the new prediction error with the higher degree polynomial), then a new degree $n = n + 1$ is considered for the next iteration.

3.3 Bifurcation detection

Bifurcations are associated in changes in the stability properties between members of the family, which can be directly inferred from the eigenvalue analysis of the *monodromy matrix* $\mathbb{M}$ [9]. The algorithm is capable of detecting such bifurcations, although not fully exploited in the performed tests as it is out of the scope of the present paper.

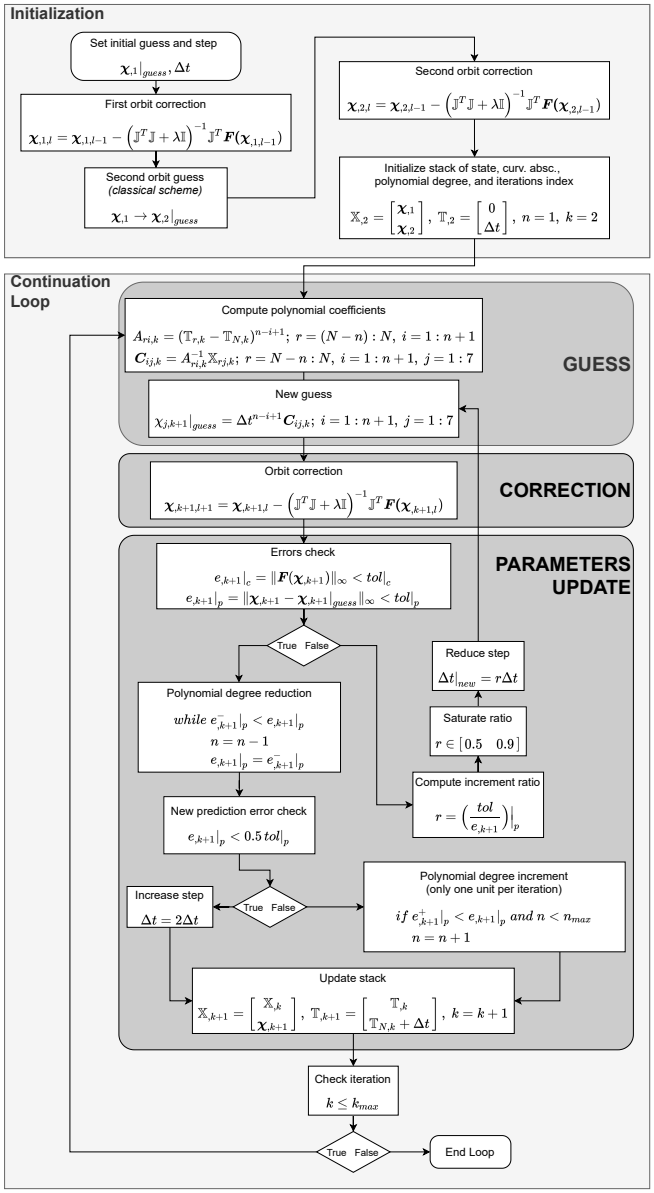


Fig. 1: Algorithm scheme

4 Results

4.1 Problem setup

Binary system parameters The algorithm is tested in the framework of the Earth-Moon binary system, considering the approximation of the CR3BP (see Section 2.1), and with the following characteristic parameters:

- $\mu = 0.0122$
- $L^* = 384\,400\,\text{km}$
- $t^* = 3.7520 \times 10^5\,\text{s}$

Initialization conditions The first two orbits, required by the proposed algorithm, are generated through a natural parameter continuation, by imposing a step along x direction (for planar trajectories) or z direction (for Halo orbits) of 1×10^{-6} in the non-dimensional units of the CR3BP. To ensure a fair comparison between the natural parameter continuation and the proposed approach, the initial curvilinear abscissa step $\bar{\Delta t}$ is set such that the corresponding first order polynomial predicts a new state with a Δx or Δz exactly matching the 1×10^{-6} step of the natural parameter approach, namely:

$$\Delta x = \bar{\Delta t}^{n-i+1} \boldsymbol{C}_{i1,1} - \boldsymbol{\chi}_{1,1} = 1 \times 10^{-6}, \quad n = 1,\, i = 1, 2; \tag{22}$$

with $\boldsymbol{C}_{i0,1}$ computed from the first two orbits according to Eq. 15.

Correction conditions Regarding the correction block of the algorithm, the selection of the Poincaré phase conditions was left open (Section 3.2.2), as it depends on the specific problem or family to be developed. For the current analysis, only symmetric orbits with respect to the x-z plane of the Earth-Moon rotating reference frame are selected. Hence one straightforward condition to ensure that the initial state $\boldsymbol{X}$ lies on such plane is:

$$\boldsymbol{\chi}_{2,k} = y_k = 0 \tag{23}$$

A second condition is required to ensure the local uniqueness of the solution: if such condition is not present, the problem is not well-constrained, and the corrected orbit may approach and overlap to the previous solution of the family. Such constraint can be obtained by imposing a fixed value on one of the other position elements of the initial node (x or z). The constrained value varies along the family, and is set to be equal to the corresponding guessed value:

$$\boldsymbol{\chi}_{1,k} = x_k = \boldsymbol{\chi}_{1,k}|_{guess} \; or \; \boldsymbol{\chi}_{3,k} = z_k = \boldsymbol{\chi}_{3,k}|_{guess} \tag{24}$$

The specific choice of fixing the x or z state depends on the initial development direction of the family. In particular, all planar orbit develop along x, hence such element is constrained. Vice versa, in the specific case of Halo orbits, z represents the main direction (as Halo generate from an out-of-plane bifurcation of Lyapunov families), hence z state is fixed.

In the present work, a correction process is considered succesful if the final residual is below 1×10^{-9}.

Continuation conditions The main parameters to be set for the analysis are the maximum degree of the polynomial (n_{max}), and the maximum number of iterations of the loop (k_{max}). The first parameter needs to be high enough to ensure that the algorithm will not be limited in speed; however, an excessively high value may introduce numerical issues, due to the high slope of the polynomial and corresponding sudden state changes even at small abscissa steps. In the present work, a maximum value of $n_{max} = 10$ is considered satisfactory for the purpose.

The selection of the maximum number of iterations is important as well as it may affect the final result performance and distort the merit parameters of the comparison. In particular, a too high number poses a limit in the computational time; vice versa, a too small value may prevent the polynomial algorithm from reaching its nominal continuation speed, making it appear less efficient than it is. A value of 100 is considered satisfactory in such sense, and it is used for all the presented results.

The continuation is considered satisfactory only if the prediction error falls below 1×10^{-4}.

Orbital families Various types of orbital families are tested to assess the capability of the algorithm to adapt to trajectories with very different properties (dimensions, shape, period, etc.). Here, such families are listed and described. The selected orbital families are:

- L2 Lyapunov Orbits (*LyL2*)
- L2 Southern Halo Orbits (*SHL2*)
- Resonant Orbit "A" (*ResA*)
- Resonant Orbit "B" (*ResB*)
- Resonant Orbit "C" (*ResC*)
- Distant Retrograde Orbits (*DRO*)

Lyapunov orbits are characterized by a planar motion around the Lagrangian points L1/L2/L3. The family starts from the respective Lagrangian point, and continues by increasing the orbit dimensions, until proximity to one of the main attractors (or both in the L1 case) is achieved. Here, L2 Lyapunov orbits are considered, however, similar conclusion can be drawn for the L1/L3 families.

Halo orbits generate as bifurcation from L1/L2 Lyapunov orbits, and develop out of the binary system's orbital plane, while approaching the Moon. In particular, orbits close to the Moon and belonging to such family are known to be particularly challenging to generate from a computational perspective [**8**], thus representing an optimal test bed for the continuation algorithms. **For each collinear Lagrangian point, a Halo family exists, continuously extending towards the out-of-plane region along both upper and lower sides. For the sake of convenience, the two branches of the family are here treated separately, and named "*northern*" and "*southern*" Halo respectively. Here, southern Halo orbits around L2 are considered.**

Resonant orbits are characterized by a specific period ratio with respect to the Moon. Depending on the ratio value, their shape greatly varies in the rotating frame of the binary system. **Their motion encompasses the whole binary system, both in and out of the system's plane depending on the specific family, alternately passing nearby the attractors and above the Earth-Moon distance [15]**. Such motion makes the dimensions of the resonant orbits significantly larger than Lyapunov Orbits, making the prediction process more relevant to ensure the convergence of the corrector.

Finally, distant retrograde orbits generate directly from the Moon, rather than from a Lagrangian point. Their shape is strongly related to their dimension, as they are distorted by Earth gravity the more their distance from the Moon increases.

Performance index The assessment of the algorithm's performance and behavior is carried out through a set of performance index, namely:

- **Pred. Err. (avg)**: prediction error, evaluated as in Eq. 20, scaled by the local abscissa step Δt, and averaged over the iterations,

$$\textbf{Pred. Err. (avg)} = \text{mean}\left[\frac{\|\boldsymbol{\chi}_{,k+1} - \boldsymbol{\chi}_{,k+1|p}\|_\infty}{\Delta t}\right] \tag{25}$$

- **No. Corr. Steps (avg)**: number of iterations from the correction inner loop, averaged over the iterations.
- **No. Reject. Steps**: total number of rejected steps within the continuation iterative process.
- **Cont. Speed (avg)**: continuation speed, computed as the 2-norm of the state difference between two consecutive solutions, averaged over the iterations,

$$\textbf{Cont. Speed (avg)} = \text{mean}\left[\|\boldsymbol{\chi}_{,k+1} - \boldsymbol{\chi}_{,k}|\|_2\right] \tag{26}$$

- **Max State Dist.**: final state "distance" achieved at the end of the continuation loop, measured as the sum of the delta-state norms of each continuation step,

$$\textbf{Max State Dist.} = \sum_{i=2}^{k_{max}} \|\boldsymbol{\chi}_{,k+1} - \boldsymbol{\chi}_{,k}|\|_2 \tag{27}$$

4.2 Algorithm performances analysis

A comparison of the proposed algorithm with classical approaches is carried out to highlight its advantages (and possible disadvantages). In particular, the following continuation schemes are analyzed:

- Natural parameter continuation,
- First degree polynomial continuation,
- Adaptive degree polynomial continuation *(proposed algorithm)*

The first method represents the easiest and most straightforward approach to the orbital families continuation problem, therefore it is evaluated to provide a reference for assessing the performance increment of the proposed new scheme. The second method is a limited version of the new proposed approach, with the polynomial degree fixed to one, to highlight the advantages of having a variable and adaptive degree. The third method is the new proposed scheme, with no limitations implemented. For sake of simplicity, the three algorithms are referred to as

- NP for natural parameter,
- FP for first degree polynomial,
- AP for adaptive degree polynomial.

The abbreviation FS and AS specifies if the algorithm exploits a fixed step or an adaptive step. Tables 1,2,3,4,5 and 6, show the full collection of the performance index for all the tested algorithms, together with the maximum degree achieved by the adaptive polynomial.

Fixed step methods By looking at the results for fixed step approaches, it can be noticed how the polynomial approaches guarantee a significantly better prediction, which consequently reduces the number of correction steps required for convergence. Nevertheless, such improvement has a minor impact on the overall algorithm. In fact, the algorithm's speed depends on both the correction process and the step length of the continuation. A better prediction slightly improves the former, but the step (which affects speed the most) is forced to be constant, posing a large limitation on the overall scheme. Therefore, the adaptivity of the step demonstrated to be a fundamental pre-condition for obtaining advantages from the polynomial prediction approach.

Adaptive step **The polynomial approaches with adaptive step show an increment in the prediction error with respect to their fixed step counterparts. This is inevitably caused by the increment of the step size (thanks to the initial smaller error), which in turn magnifies non-linearities between adjacent solutions.**

The natural parameter approach, instead, shows initial higher prediction errors, thus preventing the step from increasing. As a consequence the prediction error appears to be comparable to the fixed step natural parameter approach.

Furthermore, differences are observed between first and adaptive degree polynomials. Despite the higher continuation speed, the adaptive polynomial displays in most cases a slightly better prediction errors than the first degree one. This indicates a good capability of the adaptive polynomial of keeping the error limited while performing larger steps.

Finally, it should be noted how the maximum degree achieved by the adaptive polynomial method with adaptive step is typically higher than the fixed step approach. This is explained by the fact that larger steps introduce more non-linearities, and a first order approximation is no longer the best option.

Nevertheless, despite the degree limit has been set to 10, the maximum measured degree is equal to 3. Such low value may be justified by how the prediction tolerance is stated: the threshold value considers the absolute error, without scaling it with respect to the step; this may over-constraint the whole process, preventing the polynomial to reach higher degree values.

Exceptions There are some exceptions (*Halo, ResB* and *ResC* from Tables 2, 4, 5) where the adaptive polynomial displays an average prediction error higher than the first degree approach. Correspondingly, it is observed an increase of the average amount of rejected steps. Two different phenomena that cause this behavior are observed.

1. Concerning *ResB* and *ResC* families, the rapid increase of rejected steps and the higher average prediction error are caused by the high continuation speed of the method itself: the algorithm is capable of reaching the final orbits of the families (close to the attractors), where steeper changes in the dynamics are present; because of that, prediction errors become larger, and the algorithm is forced to slow down by repeatedly rejecting the iterations and reducing the step length. This can be observed in Fig. 2 for the *ResC* case.
2. For what concerns the Halo family, a different issue is detected: as the algorithm moves along the family, the continuation direction changes from z to x; however, the Poincaré condition continues to force the z value. The consequence is that for minor changes in z direction are associated with large along x. Although the correction process is capable of converging to the new solution, the final prediction error results much larger, preventing the step to be updated to higher values (see Fig. 3). Nevertheless such undesired condition can be easily solved by changing the Poincaré phase condition: as an example, the issue may be overcome if the distance from previous solution (i.e. $\sqrt{(x_{,k+1} - x_{,k})^2 + (z_{,k+1} - z_{,k})^2}$) is taken as new condition instead of fixing x or z.

	Pred. Err. (avg)	No. Corr. Steps (avg)	No. Reject. Steps	Cont. Speed (avg)	Max State Dist.	Max Degree
NP (FS)	5.44	7	n.a.	5.54×10^{-6}	5.57×10^{-4}	n.a.
FP (FS)	1.84×10^{-2}	4	n.a.	5.54×10^{-6}	5.57×10^{-4}	n.a.
AP (FS)	1.84×10^{-2}	4	n.a.	5.54×10^{-6}	5.57×10^{-4}	1
NP (AS)	5.47	8	0	8.71×10^{-5}	8.60×10^{-3}	n.a.
FP (AS)	44.84	5	14	4.00×10^{-3}	3.38×10^{-1}	n.a.
AP (AS)	12.98	5	20	3.26×10^{-2}	2.58	3

Table 1: Lyapunov continuation performances.

	Pred. Err. (avg)	No. Corr. Steps (avg)	No. Reject. Steps	Cont. Speed (avg)	Max State Dist.	Max Degree
NP (FS)	5.31×10^{-2}	4	n.a.	1.00×10^{-6}	1.00×10^{-4}	n.a.
FP (FS)	1.96×10^{-2}	2	n.a.	1.00×10^{-6}	1.00×10^{-4}	n.a.
AP (FS)	1.96×10^{-2}	2	n.a.	1.00×10^{-6}	1.00×10^{-4}	1
NP (AS)	2.95×10^{-1}	6	15	3.36×10^{-4}	2.82×10^{-2}	n.a.
FP (AS)	57.78	5	7	3.20×10^{-3}	2.95×10^{-1}	n.a.
AP (AS)	75.84	4	28	1.18×10^{-2}	8.37×10^{-1}	2

Table 2: Halo continuation performances.

	Pred. Err. (avg)	No. Corr. Steps (avg)	No. Reject. Steps	Cont. Speed (avg)	Max State Dist.	Max Degree
NP (FS)	6.26	5	n.a.	6.45×10^{-6}	6.47×10^{-4}	n.a.
FP (FS)	7.30×10^{-3}	3	n.a.	6.45×10^{-6}	6.47×10^{-4}	n.a.
AP (FS)	7.30×10^{-3}	3	n.a.	6.45×10^{-6}	6.47×10^{-4}	1
NP (AS)	6.25	6	0	6.96×10^{-5}	6.90×10^{-3}	n.a.
FP (AS)	12.54	6	1	8.40×10^{-3}	8.20×10^{-1}	n.a.
AP (AS)	9.43	22	39	5.70×10^{-2}	3.42	3

Table 3: Resonant A continuation performances.

	Pred. Err. (avg)	No. Corr. Steps (avg)	No. Reject. Steps	Cont. Speed (avg)	Max State Dist.	Max Degree
NP (FS)	2.30	5	n.a.	2.91×10^{-6}	2.90×10^{-4}	n.a.
FP (FS)	3.00×10^{-3}	3	n.a.	2.91×10^{-6}	2.90×10^{-4}	n.a.
AP (FS)	3.00×10^{-3}	3	n.a.	2.91×10^{-6}	2.90×10^{-4}	1
NP (AS)	2.30	7	0	9.01×10^{-5}	8.90×10^{-3}	n.a.
FP (AS)	14.37	6	6	8.10×10^{-3}	7.51×10^{-1}	n.a.
AP (AS)	75.42	28	37	2.39×10^{-2}	1.48	3

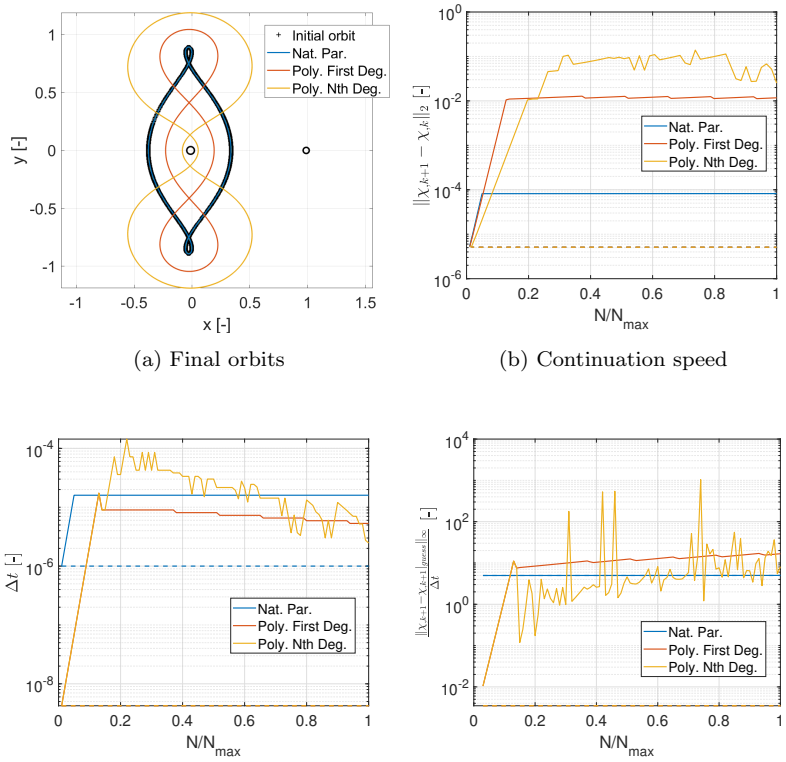
Table 4: Resonant B continuation performances.

	Pred. Err. (avg)	No. Corr. Steps (avg)	No. Reject. Steps	Cont. Speed (avg)	Max State Dist.	Max Degree
NP (FS)	4.98	5	n.a.	5.11×10^{-6}	5.12×10^{-4}	n.a.
FP (FS)	3.50×10^{-3}	3	n.a.	5.11×10^{-6}	5.12×10^{-4}	n.a.
AP (FS)	3.50×10^{-3}	3	n.a.	5.11×10^{-6}	5.12×10^{-4}	1
NP (AS)	4.99	6	0	8.02×10^{-5}	7.90×10^{-3}	n.a.
FP (AS)	11.26	6	6	1.07×10^{-2}	9.96×10^{-1}	n.a.
AP (AS)	29.72	24	39	6.01×10^{-2}	3.61	3

Table 5: Resonant C continuation performances.

	Pred. Err. (avg)	No. Corr. Steps (avg)	No. Reject. Steps	Cont. Speed (avg)	Max State Dist.	Max Degree
NP (FS)	14.74	6	n.a.	1.63×10^{-5}	1.60×10^{-3}	n.a.
FP (FS)	5.79×10^{-1}	4	n.a.	1.63×10^{-5}	1.60×10^{-3}	n.a.
AP (FS)	5.79×10^{-1}	4	n.a.	1.63×10^{-5}	1.60×10^{-3}	2
NP (AS)	14.76	6	0	6.48×10^{-5}	6.40×10^{-3}	n.a.
FP (AS)	137.02	5	3	1.55×10^{-2}	1.49	n.a.
AP (AS)	34.99	5	10	5.94×10^{-2}	5.29	2

Table 6: Distant Retrograde Orbits continuation performances.



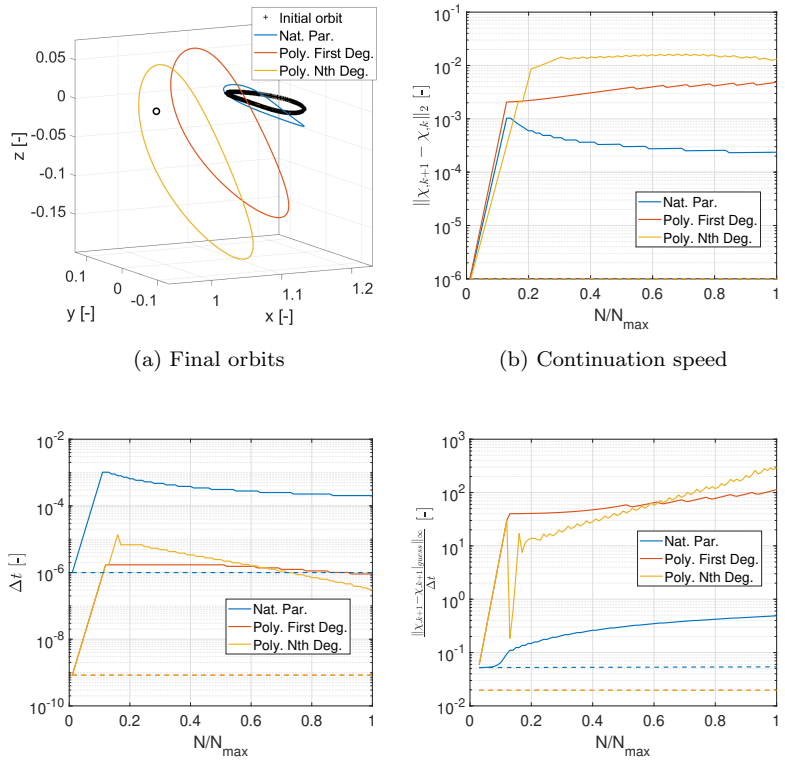
(a) Final orbits

(b) Continuation speed

(c) Step length

(d) Prediction error

Fig. 2: Res C continuation performance. Solid lines represent the adaptive step methods, while dashed lines indicate the corresponding fixed step methods.



(a) Final orbits

(b) Continuation speed

(c) Step length

(d) Prediction error

Fig. 3: SHL2 continuation performance. Solid lines represent the adaptive step methods, while dashed lines indicate the corresponding fixed step methods.

5 Conclusions

In the present paper, a continuation scheme for generating periodic orbits in the CRTBP, leveraging adaptive-degree polynomial prediction functions, is proposed and compared to two classical schemes, natural parameter and first order methods. The purpose of the new algorithm is to improve the overall performance of the continuation process, by speeding up the motion along the development directions of the family. The objective is achieved thanks to the improved prediction performances, capable of capturing the nonlinearities and adapting to them (through the variable degree) along the search space, thus ensuring larger search steps without hindering the convergence of the correction algorithm.

A comparison with a fixed step approach has been carried out, to highlight the strong correlation between degree and step adaptivity. In particular, it has been observed that the adaptive-degree polynomial outperforms first order or natural parameter continuation schemes when associated with an adaptive

step, while negligible improvements are present when fixed step is employed, due to the great importance of this parameter on the overall speed of the process.

Some minor issues in the continuation process were observed for specific orbital families. In particular, the continuation tolerance demonstrated to be a critical parameter for the algortihm speed, as it greatly prevented the degree of the polynomial from rising. The main reason of this limitation was attributed to the absolute nature of the continuation error threshold, easily exceeded when the step is large. A possible workaround is then proposed, by employing a relative tolerance, scaled over the local step. A second issue, related to Poincaré phase conditions, was observed. The simple "fixed state elements" constraints demonstrated to be acceptable for planar orbits, but prevented the algorithm from achieving higher speed where more complex orbits are explored (as for Halo orbits). A different formulation of such constraints (e.g. position distance from previous solution) may be employed to make the process independent from the continuation direction.

References

1. Allgower, E.L., Georg, K.: Numerical path following. In: Techniques of Scientific Computing (Part 2), *Handbook of Numerical Analysis*, vol. 5, pp. 3 – 207. Elsevier (1997). DOI https://doi.org/10.1016/S1570-8659(97)80002-6. URL `http://www.sciencedirect.com/science/article/pii/S1570865997800026`
2. Baresi, N., Fu, X., Armellin, R.: A high-order taylor polynomials approach for continuing trajectories in restricted three-body problems. In: Proceedings of the 2020 AAS/AIAA Astrodynamics Specialist Conference (2020)
3. Dichmann, D., Doedel, E., Paffenroth, R.: The computation of periodic solutions of the 3-body problem using the numerical continuation software auto. In: Libration Point Orbits and Applications, pp. 489–528. World Scientific (2003)
4. Doedel, E.J., Paffenroth, R.C., Keller, H.B., Dichmann, D., Galán-Vioque, J., Vanderbauwhede, A.: Computation of periodic solutions of conservative systems with application to the 3-body problem. International Journal of Bifurcation and Chaos **13**(06), 1353–1381 (2003)
5. Doedel, E.J., Romanov, V.A., Paffenroth, R.C., Keller, H.B., Dichmann, D.J., Galán-Vioque, J., Vanderbauwhede, A.: Elemental periodic orbits associated with the libration points in the circular restricted 3-body problem. International Journal of Bifurcation and Chaos **17**(08), 2625–2677 (2007)
6. Eugene L. Allgower, K.G.: Introduction to numerical continuation methods. Classics in Applied Mathematics. Society for Industrial Mathematics (1987). URL `http://gen.lib.rus.ec/book/index.php?md5=5fd3238192cccfb159f35916a6b31f75`
7. Ferrari, F.: Non-keplerian models for mission analysis scenarios about small solar system bodies. Ph.D. thesis (2016)
8. Howell, K., Breakwell, J.: Almost rectilinear halo orbits. Celestial mechanics **32**(1), 29–52 (1984)
9. Howell, K.C.: Three-Dimensional Periodic Halo Orbits. Celestial Mechanics **32**(1), 53 (1984)
10. Lundberg Bruce N.; Poore, A.B.: Variable order adams-bashforth predictors with an error-stepsize control for continuation methods. SIAM Journal on Scientific and Statistical Computing **12**, 695–723 (1991). DOI 10.1137/0912037. URL `http://doi.org/10.1137/0912037`
11. Schwetlick Hubert; Cleve, J.: Higher order predictors and adaptive steplength control in path following algorithms. SIAM Journal on Numerical Analysis **24**, 1382–1393 (1987). DOI 10.1137/0724089. URL `http://doi.org/10.1137/0724089`

12. Shampine, L.F.: Computer solution of ordinary differential equations. The Initial Value Problem (1975)
13. Stoer, J., Bulirsch, R.: Introduction to numerical analysis. Texts in applied mathematics. Springer (2002)
14. Szebehely, V.: Theory of Orbits: The Restricted Problem of Three Bodies. Academic Press, New York and London (1967)
15. Vaquero, M., Howell, K.C.: Leveraging resonant-orbit manifolds to design transfers between libration-point orbits. Journal of Guidance, Control, and Dynamics **37**(4), 1143–1157 (2014)
16. Zimovan-Spreen, E.M., Howell, K.C., Davis, D.C.: Near rectilinear halo orbits and nearby higher-period dynamical structures: orbital stability and resonance properties. Celestial Mechanics and Dynamical Astronomy **132**(5), 1–25 (2020)