2023

Trajectory Design Optimization Using Coupled Radial Basis Functions (CRBFs)

Kyler Roy
University of Central Florida

TRAJECTORY DESIGN OPTIMIZATION USING COUPLED RADIAL BASIS FUNCTIONS (CRBFS)

by

KYLER AUSTIN ROY
B.S. University of Central Florida, 2021

A thesis submitted in partial fulfillment of the requirements
for the degree of Master of Science
in the Department of Mechanical and Aerospace Engineering
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Summer Term
2023

ABSTRACT

Optimal trajectory design has been extensively studied across multiple disciplines adopting different techniques for implementation and execution. It has been utilized in past space trajectory missions to either optimize the amount of fuel spent or minimize the time of flight to meet mission requirements. Coupled Radial Basis Functions (CRBFs) are a new way to solve these optimal control problems, and this thesis applies CRBFs to spacecraft trajectory optimization design problems. CRBFs are real-valued radial basis functions (RBFs) that utilize a conical spline while also not being affected by the value of the shape parameter. The CRBF approach is applied to nonlinear optimal control problems. We adopt the indirect formulation so that the necessary and boundary conditions are derived from the system dynamical equations. As a result, a set of nonlinear algebraic equations (NAEs) is generated. The NAEs are then solved using a standard solver in MATLAB and the results are produced. CRBFs do not rely heavily on initial extensive analysis of the problem, which makes it very intuitive to use. The states, control, and co-states are defined as the equations to be solved and approximated using CRBFs. The results show that CRBFs can be applied to space trajectory optimization problems to produce accurate results across state and costate variables on uniform user defined nodes across the simulation time.

ACKNOWLEDGMENTS

This work would not have been done without the assistance of Dr. Tarek A. Elgohary and Dr. Ahmed E. Seleit. Their continuous support and guidance through education and application was above expectation and created an enjoyable experience while also exceedingly advancing my technical knowledge and communication skills in an area of study I have always been interested in.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

Acronym	Definition
BVP	Boundary Value Problem
CRBF	Coupled Radial Basis Function
ICLOCS2	Imperial College London Optimal Control Software
IVP	Initial Value Problem
NAE	Nonlinear Algebraic Equation
RBF	Radial Basis Function
UAV	Unmanned Aerial Vehicle

CHAPTER ONE: INTRODUCTION

The optimal control theory has been around since the 20^{th} century with some of the most significant discoveries happening in the 1940s and 1950s. A subfield of this is optimal trajectory design which focuses on finding an optimal path or trajectory that a system must follow in order to reach an objective. Optimal trajectory design has various applications throughout different fields including, but not limited to: aerospace engineering, robotics, control systems, guidance and navigation, manufacturing and process control, and biomechanics. In aerospace engineering, optimal trajectory design plays a crucial role in space missions, satellite deployments, and planetary transfers. While considering numerous constraints such as fuel consumption, time, and mission objectives, optimal trajectory design helps determine the most efficient and cost-effective paths for spacecraft to take.

Much of the foundation for optimal trajectory design was set by mathematicians such as Pierre-Louis Lions, Lev Pontryagin, and Richard Bellman in the 1940s and 1950s. Some of the concepts they developed were the maximum principle and dynamic programming. During the mid-20^{th} century, optimal trajectory design was noticed by the space community and began implementing different solutions into distinctive missions such as the Apollo mission. With advancements in computing power and numerical methods happening in the 1960s and 1970s, practical techniques were developed. These techniques include direct methods, indirect methods, shooting methods, collocation methods, and pseudospectral methods. As optimal trajectory problems became more complex through the 1980s, advancements in direct and collocation methods began to emerge. Current algorithms and machine learning have been an active area of study when being applied to high dimensional systems and real time planning for self-sufficient systems.

Radial Basis Function (RBF) collocation has been around for a few decades in solving the optimal trajectory design problem, but still remain as a trustworthy method. RBFs are mathematical functions used for approximating the trajectory of the system. While being a very efficient method in approximating nonlinear dynamic equations, RBFs stumble into an issue when implementing the shape parameter. This work aims to solve this issue with a new method called Coupled Radial Basis Functions (CRBFs) in which a conical spline is utilized to solve the optimal trajectory design problem. As this method has been used to solve benchmark optimal control problems, it has not been utilized in the optimal trajectory design field. Despite the current advancements in solving the optimal trajectory design problem and recent implementation of RBFs, understanding how they can be improved and involve less analytical analysis remains important. The purpose of this study is to advance the research field with another way to solve the optimal trajectory problem.

Firstly, shooting methods will employed in MATLAB to solve multiple orbit trajectory design problems. Subsequently, the new CRBF method will be used in MATLAB as well. The results between the two methods will be compared to show accuracy of the CRBF method. These results will include the path of the spacecraft along with the states and costates over time. The significance in applying CRBFs to trajectory design problems will allow the community to establish a new understanding and advancement in the aerospace engineering field for solving these problems with less analytical background while also not relying on the shape parameter.

The thesis is structured as follows: in Chapter 2, a review of all the recent and past literature is done to show to significance of this study. In Chapter 3, the methodology of applying CRBFs to the optimal trajectory design problem is portrayed as well as the development of the optimal trajectory problems being solved. In Chapter 4, the findings of the shooting method and CRBFs are discussed and compared. Lastly, in Chapter 5, the conclusions are shown.

CHAPTER TWO: LITERATURE REVIEW

The purpose of this literature review is to provide a comprehensive analysis of the existing research on the techniques and methods developed in solving the optimal control problem. With the advancement and execution of past methods, current systems have evolved to be more complex which rely heavily on both modern-day technology and extensive analyses of the situation. This literature review aims to evaluate the most recent body of knowledge, identify key themes, research gaps, and methods used in previous studies. By evaluating the various perspectives, this review pursues to summarize the past findings in solving the optimal control problem and identify how this thesis can fill in those research gaps.

Many techniques have been explored to solve the optimal control problem. Stryk and Bulirsch (1992) explain the differences between direct and indirect methods. While indirect methods are based on the calculus of variations, also known as the maximum principle, direct methods transform the optimal control problem into a system of NAEs. They point out that the user must have a well-established understanding of the problem in order to define its equations correctly. Direct methods lack in producing accurate results as compared to indirect methods due to the complicated nature of the problem. They further went into detail about a hybrid approach by combining the convergence properties of the direct method and coupling it with the reliability and accuracy of indirect methods. This method was successfully applied to many test cases, but further improvement is needed for approximation of the control variables. Osborne (1969) explains that shooting methods issues are caused by instabilities of the initial value problem as well as having the proper starting values. Though this technique may be widely available on most computers, it still relies heavily on manual analysis or guesses in order to implement a starting point.

Figure 1 shows the break out between indirect and direct methods and how they correlate. Depending on the type of simulation and how often the dynamics are integrated will decipher which numerical method will be used to solve the optimal control problem. This study focuses on an indirect collocation method highlighted by the red flow.

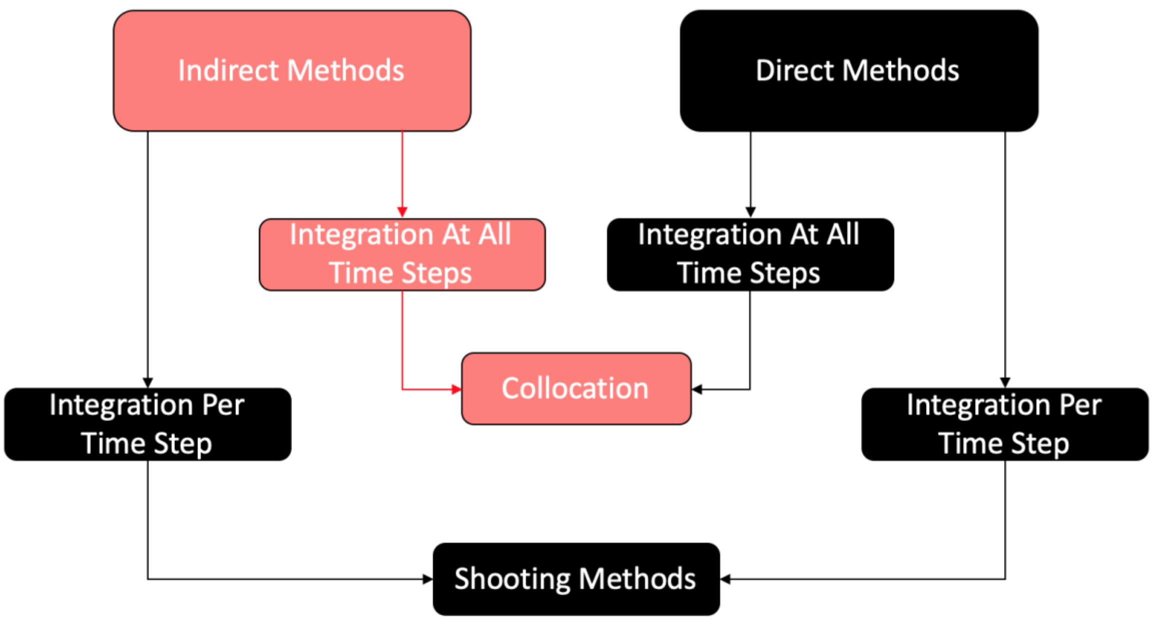

Figure 1 Indirect and Direct Method Flow Diagram

Cots et al. (2016) utilize both indirect and direct methods in solving the minimum time and minimum fuel consumption problem of a climbing aircraft. The indirect method utilizes the maximum principle with state constraints of the problem and is transformed into a boundary value problem (BVP). The problem is eventually initialized using the direct method to find an optimal initial guess and structure of the problem. Rao (2010) goes into deep detail on the pros and cons of both indirect methods and direct methods. While indirect methods have an advantage in regards to simplicity and highly accurate solutions, it is extremely sensitive to unknown boundary conditions and requires the derivation of the first-order optimality conditions. Indirect methods lack in the ability to solve very complex optimal control problems due to this required derivation. Direct methods prove to be accurate and robust in cases where the control can be parametrized simply and the problem can be characterized accurately. Highly complex optimal control problems can easily be formulated and solved using NLP solvers, due to the nature of them not requiring a good initial guess. Direct methods are also computationally efficient due to being able to exploit the sparsity in the derivatives of the constraints and objective function. Betts (1998) explains that indirect methods requires analytically computing the gradient, then utilizing a root-finding algorithm that finds the set of variables where the gradient is zero. This root-finding algorithm has a small area of convergence due to the initial guesses possibly not having a good meaning of physical interpretation. Betts then explains that direct methods does not require initial guesses for the adjoint variables since the states and control variables are adjusted to directly solve the optimal control problem. Another note is that parameterization of the control variables is utilized during direct methods. Bianco et al. (2018) compares direct and indirect methods in solving the minimum lap time problems. While both methods displayed similar behavior, the implementation features of each had some differences including the integration scheme, and the projection of the state.

Huntington (2008) compared global and local collocation. Global collocation implements a global polynomial across the entire time interval of the mission, while local collocation discretizes the problem into certain points so that it's able to support the behavior of the dynamics at that time. Again, these methods still rely heavily on initial knowledge of the problem. There have been many different types of discretization techniques used in local collocation including the pseudospectral method which utilizes Legendre-Gauss-Lobatto points to collocate the system dynamics (Elnagar et al., 1995). Narayanaswamy and Damaren (2020) implemented the Legendre-Gauss and Hermite-Legendre-Gauss-Lobatto direct collocation methods for a minimum-time low thrust mission in which utilized the Edelbaum trajectory to create an initial guess. A common theme among local and global collocation is developed such that there must be a heavy physical and analytical understanding of the problem in order to define proper initial conditions. Improper initial conditions can lead to divergence among these methods studied.

A Radial Basis Function (RBF) is a real-valued function in which its value is dependent on the distance from a fixed point. Kansa (1990) introduced RBFs and showed that they are more efficient and effective than standard finite difference approximations. Mirinejad and Inanc (2017) introduced an RBF collocation method for solving optimal control problems. They show that is more accurate and effective when compared to a local polynomial method and a global polynomial method when solving an unmanned aerial vehicle (UAV) navigation problem. RBFs have been widely used to solve differential equations across many different applications. Tatari and Dehghan (2010) apply RBFs to the heat equation. Kazem et al. (2012) use the RBF method for solving the Fokker–Planck equation. Tolstykh and Shirobokov (2003) utilize RBF approximation techniques to solve non-linear Karman-Fopple equations. Finally, Carr et al. (1997) employ RBFs on interpolating incomplete surfaces for medical imaging. For infinitely smooth RBFs, a shape parameter must be implemented to tune the shape of the RBF. Increasing and decreasing this parameter effects the RBF's shape drastically. Finding the optimal shape parameter can be complicated and involves excessive calculations as a part of the methodology of RBFs. Previous research does try and implement a strategy of selecting the best shape parameter, but none have tried to remove the shape parameter all together. For example, Bhatia and Arora (2016) describe that it is difficult to find the best shape parameter for various RBFs. Utilizing a small shape parameter gives the best result for some RBFs, but the interpolation matrix becomes ill-conditioned. Mongillo (2011) tested the properties of the shape parameter and tried to create a method for finding an optimal shape parameter. The potential instability in the proposed methods may risk inaccurate results to be produced. Koupaei et al. (2018) produced a new algorithm to find the optimal shape parameter, but as the domain of the problem increased, the accuracy of the algorithm decreased. Chen et al. (2023) studied correlations between finding a "good" shape parameter and the effective condition number. Again, choosing the shape parameter played a vital role in determining the error in RBF approximation.

RBF collocation has been heavily studied in solving different types of problems. Elgohary et al. (2014a) presents an RBF collocation algorithm that obtains a solution for the initial value problem (IVP). Elgohary et al. (2014b) also used RBF collocation to solve the Duffing optimal control problem with initial and final conditions as well as the orbital transfer Lambert's problem. Elgohary (2015) then utilized RBFs combined with time collocation to produce an integrator that's able to handle orbit propagation problems. This deep research has shown that

RBF collocation is a valid technique to solve many different problems. Seleit (2022) introduced a new method called coupled radial basis functions (CRBFs) which couples traditional RBFs with a conical spline. His study created and developed this new method for benchmark optimal control problems including Zarmelo's problem, the nonlinear Duffing oscillator problem, and the nonlinear cart pendulum problem. Seleit validated the CRBF collocation method using the shooting method and the Imperial College London Optimal Control Software (ICLOCS2) and proved the accuracy against the exact solution. Kelly et al. (2022) created a MATLAB tool for CRBF collocation techniques to solve nonlinear optimal control problems.

This thesis aims to fill that gap in optimal trajectory design methods by applying the CRBF method to nonlinear spacecraft trajectory optimization problems and eliminating the selection of the shape parameter all together.

CHAPTER THREE: METHODOLOGY

This research is designed to implement a new method for solving optimal trajectory problems that display insensitivity to the shape parameter. This involves applying a theoretical and computational approach already designed and employed on benchmark optimal control problems such as Zarelmo's problem, Duffing oscillator problem, and cart problem (Seleit, 2022). This thesis focuses on applying this new approach to the maximum orbit radius and thrust control problems in spacecraft trajectory design.

No participants or data collection methods are used for this study as it only involves computational methods, simulated data, and mathematical models for real-world application.

CRBF Implementation In Optimal Control

Generally, the continuous-time optimal control problem consists of a cost functional subject to a set of dynamic equations and both path and endpoint conditions. Bryson and Ho (1975) presents the methodology to solve the optimal control problem. The continuous-time cost function to be minimized is written as

$$J = \Phi\big(x(t_f), t_f\big) + \int_{t_0}^{t_f} \mathcal{L}(x(t), u(t), t)dt \qquad (1)$$

where Φ is the endpoint or terminal cost and also referred to as the Mayer Term (Soler & Hansen, 2014) and $\mathcal{L}$ is the Lagrangian term or the running cost, which encompasses the states and control throughout the time interval of minimization. The cost functional, J, is subject to the following dynamics and boundary conditions respectively which rely heavily on the system and environments being evaluated

$$\dot{x}(t) = f(x(t), u(t), t) \tag{2}$$

$$g\big(x(t_0), t_0, x(t_f), t_f\big) = 0 \tag{3}$$

The dynamics are the main definition of the system that is being optimized and is critical that it is defined clearly and accurately. The boundary conditions encompass both path and endpoint conditions. Generally, path constraints are in turn inequality conditions that may not be functioning during the optimality of the solution. In this work, path constraints will not be studied. Optimal control problems are nonlinear which adds complexity to numerically solving them. The Lagrange multipliers are used to obtain the augmented performance index, J_a

$$J_a = \Phi\big(x(t_f), t_f\big) + \int_{t_0}^{t_f} \mathcal{L}(x(t), u(t), t) + \lambda^T(t)(f\big(x(t), u(t), t - \dot{x}(t)\big)dt \tag{4}$$

The Hamiltonian function is defined to solve optimal control problems numerically

$$\mathcal{H} = \mathcal{L}(x(t), u(t), t) + \lambda^T(t) f(x(t), u(t), t) \tag{5}$$

The augmented performance index is re-written as

$$J_a = \Phi\big(x(t_f), t_f\big) + \int_{t_0}^{t_f} \mathcal{H} - \lambda^T(t)(f\big(\dot{x}(t)\big)dt \tag{6}$$

In terms of x, u, and λ, the variation of the augmented performance index, δJ_a is expressed as

$$\delta J_a = \Phi_x^T \delta x|_{t_f} + \int_{t_0}^{t_f} [\mathcal{H}_x^T \delta x - \lambda^T(t)\delta\big(\dot{x}(t)\big) + \mathcal{H}_\lambda^T \delta\lambda - \big(\dot{x}^T(t)\big)\delta\lambda + \mathcal{H}_u^T \delta u]\, dt = 0 \quad (7)$$

Equation (7) is re-written as the following where $(\,)_* = \frac{\partial()}{\partial *}$

$$\delta J_a = (\Phi_x - \lambda)^T \delta x|_{t_f} + \lambda^T \delta x|_{t_0}$$

$$+ \int_{t_0}^{t_f} [\big(\mathcal{H}_x + \dot{\lambda}(t)\big)^T \delta x + (\mathcal{H}_\lambda + \dot{x}(t))^T \delta\lambda + \mathcal{H}_u^T \delta u]\, dt = 0 \quad (8)$$

The constant variance over time allows the removal of $\int_{t_0}^{t_f} [\big(\mathcal{H}_x + \dot{\lambda}(t)\big)^T \delta x + (\mathcal{H}_\lambda + \dot{x}(t))^T \delta\lambda + \mathcal{H}_u^T \delta u]\, dt$ and leads to the following necessary conditions or Euler-Lagrange equations

$$\dot{x} = \frac{\partial \mathcal{H}}{\partial \lambda} = f(x, u, t) \quad (9)$$

$$\dot{\lambda} = -\frac{\partial \mathcal{H}}{\partial x} \quad (10)$$

$$\frac{\partial \mathcal{H}}{\partial u} = 0 \quad (11)$$

The costate variables, λ, are referred to as the Lagrange multipliers which are tied to the state equations. The methodology for applying CRBFs to solve the optimal control problem is:

- Formulate the optimal control problem based off the certain mission

- Derive the necessary conditions from the Hamiltonian function
- Utilize CRBFs to approximate the states, costates, and control in the two-point boundary value problem

This transformation creates a set of nonlinear equations, which are solved using a nonlinear solver in MATLAB. The approximation of the states, costates, and control is written as the following with φ being the approximation matrix.

$$x(t) \cong \bar{x}(r) = \sum_{i=1}^{N} \alpha_i \varphi_i(r) \tag{12}$$

$$\lambda(t) \cong \bar{\lambda}(r) = \sum_{i=1}^{N} \beta_i \varphi_i(r) \tag{13}$$

$$u(t) \cong \bar{u}(r) = \sum_{i=1}^{N} \gamma_i \varphi_i(r) \tag{14}$$

where N is the number of collocation points and α, β, and γ are coefficient vectors. The size of φ is altered depending on the size of N. φ is approximated as follows

$$\varphi = \begin{bmatrix} \varphi(\|t_1 - t_1\|) & \varphi(\|t_2 - t_1\|) & \cdots & \varphi(\|t_N - t_1\|) \\ \varphi(\|t_1 - t_2\|) & \varphi(\|t_2 - t_2\|) & \cdots & \varphi(\|t_N - t_2\|) \\ \vdots & \vdots & \vdots & \vdots \\ \varphi(\|t_1 - t_N\|) & \varphi(\|t_2 - t_N\|) & \cdots & \varphi(\|t_N - t_N\|) \end{bmatrix} \tag{15}$$

The approximation function, or CRBF, is defined as the following in Table 1.

Table 1 Common CRBF Definitions

Type	$\varphi(r,c)$, where $r = \|t(N) - t_N\|$, $h = \frac{r}{c}$, $F = \sqrt{h^2+1}$
Coupled Multiquadric	$F + r^5$
Coupled Gaussian	$e^{h^2} + r^5$
Coupled Inverse Multiquadric	$1/F^2 + r^5$

Note that implementing the conical spline, r^5, allows the user to utilize a wide range of shape parameters, c. A higher shape parameter will not affect the effectivity of the CRBF due to the conical spline. Classical RBFs do not have this conical spline, which in turn will cause an unwanted unstable behavior.

Carl Runge in the early 20th century observed what is now called Runge's phenomenon. This refers to the poor approximation that can occur when using polynomial interpolation, especially with equally spaced interpolation points. Runge's phenomenon can manifest upon certain functions that are highly oscillatory or have sharp variations. Figure 2 shows two different types of nodal distributions or interpolation points. Runge's phenomenon is mostly observed with uniformly spaced nodes. Chebyshev nodal distribution helps minimize the oscillations by non-uniformly distributing the nodes towards the boundaries unlike the uniform distribution.

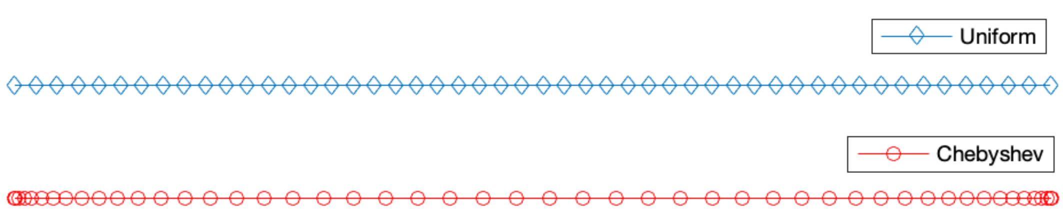

Figure 2 Nodal Distributions

Figure 3 shows *polyfit* via MATLAB, which is being plotted against a uniform nodal distribution. The function $f(x) = \frac{1}{1+25x^2}$ is being utilized to show how having a uniform distribution of the data points can affect the approximation of the problem. Notice how the polyfit plot experiences Runge's phenomenon due to the lack of nodes towards the boundaries. CRBFs help stay away from experiencing Runge's phenomenon by being able to approximate the function by utilizing any type of nodal distribution.

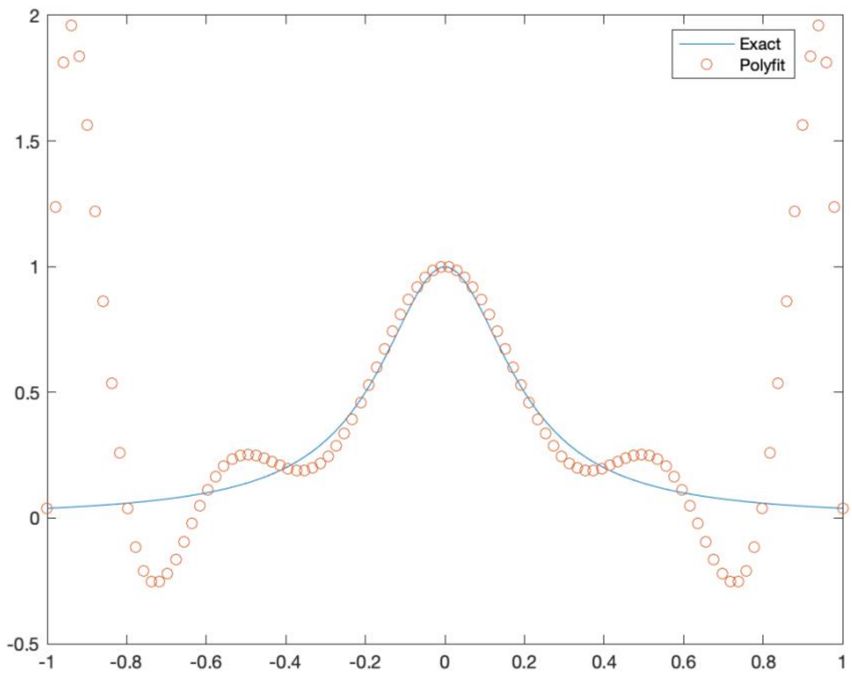

Figure 3 MATLAB Polyfit Over Uniform Distribution

Taking the time-derivatives of the approximations of the states and costates

$$\dot{x}(r) = \sum_{i=1}^{N} \alpha_i \dot{\varphi}_i(r) \quad (16)$$

$$\dot{\lambda}(r) = \sum_{i=1}^{N} \beta_i \dot{\varphi}_i(r) \quad (17)$$

To better represent these approximations, a new matrix is defined, D, as follows

$$\alpha = \varphi^{-1}\bar{x} \tag{18}$$

$$\dot{\bar{x}} = \alpha\dot{\varphi} \tag{19}$$

$$\dot{\bar{x}} = \varphi^{-1}\bar{x}\dot{\varphi} \tag{20}$$

$$\dot{\bar{x}} = D\bar{x} \tag{21}$$

The approximate set of NAEs is written as

$$\dot{x} \cong \dot{\bar{x}} = D\bar{x} = \frac{\partial\mathcal{H}}{\partial\bar{\lambda}} \tag{22}$$

$$-\dot{\lambda} \cong \dot{\bar{\lambda}} = -D\bar{\lambda} = \frac{\partial\mathcal{H}}{\partial\bar{x}} \tag{23}$$

$$0 \cong \frac{\partial\mathcal{H}}{\partial\bar{u}} \tag{24}$$

These are solved using an ordinary nonlinear algebraic equation (NAE) solver in MATLAB.

In summary, the implementation of the proposed method involved the following steps:

1. Defining the discretization of the system by establishing the number of collocation points, equations, and instituting the simulation time defined for the problem. As described in Table 1 the CRBF type is also selected in this structure and passed to the system parameter function.

2. The system parameters are collected and stored in a different structure which is eventually passed to the dynamics of the system.
3. Using the number of collocation points, simulation time, and CRBF type the approximation matrices are created and stored in another structure which are used to approximate the states and costates.
4. The main portion of the work is established in defining the dynamics of the system. Along with this portion of the code, the boundary and necessary conditions are defined. These functions are solved using MATLAB fsolve or fmincon (nonlinear solvers) and the solution is created.
5. The solution goes through some post processing to remove any repetitions and is cleaned up to eventually produce the results and trajectory of the spacecraft.

Maximum Radius Orbit Transfer Problem

The maximum radius orbit transfer problem involves getting a spacecraft from a fixed initial state to a free final state where the final radius is unknown. This certain mission involves a fixed time of 193 days between an Earth to Mars orbit transfer with fuel consumption being accounted for. The general CRBF approach defined is applied to the maximum radius orbit transfer problem in a given time (Bryson & Ho, 1975).

The visual representation of this problem is shown in Figure 4.

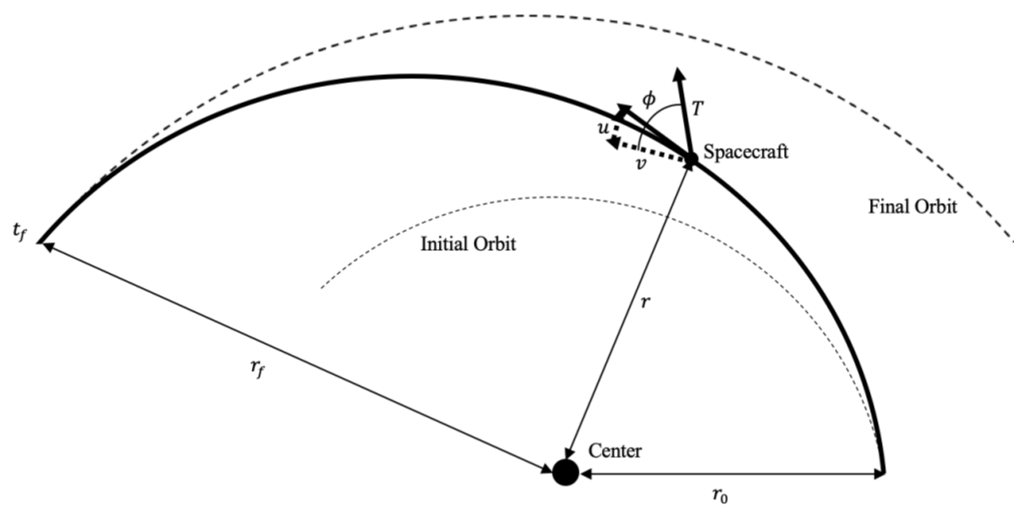

Figure 4 Maximum Radius Orbit Transfer in a Given Time

Source: Bryson and Ho, 1975, p. 66

Using the classification of the parameters, the dynamics of the problem are written as:

$$\dot{r} = u \quad (25)$$

$$\dot{u} = \frac{v^2}{r} - \frac{\mu}{r^2} + \frac{T sin\phi}{m_0 - |\dot{m}|t} \quad (26)$$

$$\dot{v} = -\frac{uv}{r} + \frac{T cos\phi}{m_0 - |\dot{m}|t} \quad (27)$$

The Hamiltonian is defined as

$$H = \lambda_r u + \lambda_u \left(\frac{v^2}{r} - \frac{\mu}{r^2} + \frac{T sin\phi}{m_0 - |\dot{m}|t}\right) + \lambda_v \left(-\frac{uv}{r} + \frac{T cos\phi}{m_0 - |\dot{m}|t}\right) \quad (28)$$

with

$$\Phi = r(t_f) \quad (29)$$

Taking the partial derivatives of the Hamiltonian with respect to the states and control

$$\dot{\lambda}_r = -\frac{\partial H}{\partial r} = -\lambda_u\left(-\frac{v^2}{r^2} + \frac{2\mu}{r^3}\right) - \lambda_v\left(\frac{uv}{r^2}\right) \quad (30)$$

$$\dot{\lambda}_u = -\frac{\partial H}{\partial u} = -\lambda_r + \lambda_v\frac{v}{r} \quad (31)$$

$$\dot{\lambda}_v = -\frac{\partial H}{\partial v} = -\lambda_u\left(\frac{2v}{r}\right) + \lambda_v\left(\frac{u}{r}\right) \quad (32)$$

$$0 = \frac{\partial H}{\partial \phi} = (\lambda_u cos\phi - \lambda_v sin\phi)\frac{T}{m_0 - |\dot{m}|t} \rightarrow tan\phi = \frac{\lambda_u}{\lambda_v} \rightarrow \phi = \tan^{-1}(\frac{\lambda_u}{\lambda_v}) \quad (33)$$

The known boundary conditions at the initial and final time are written as

$$r(t_0) = 4.9081e^{11}\, ft \quad (34)$$

$$u(t_0) = 0 \quad (35)$$

$$v(t_0) = \sqrt{\frac{\mu}{r_0}} \quad (36)$$

$$\lambda_r(t_f) = \frac{\partial \Phi}{\partial r} = 1 \quad (37)$$

$$u(t_f) = 0 \quad (38)$$

$$v(t_f) = \sqrt{\frac{\mu}{r_f}} \quad (39)$$

After defining the dynamics, costate derivatives, and boundary conditions, the residuals are expressed using Equations (22) and (23)

$$R_{1:N} = D\bar{r} - \bar{u} \quad (40)$$

$$R_{N+1:2N} = D\bar{u} - \frac{\bar{v}^2}{\bar{r}} + \frac{\mu}{\bar{r}^2} - \frac{Tsin\phi}{m_0 - |\dot{m}|t} \quad (41)$$

$$R_{2N+1:3N} = D\bar{v} + \frac{\bar{u}\bar{v}}{\bar{r}} - \frac{T cos\phi}{m_0 - |\dot{m}|t} \tag{42}$$

$$R_{3N+1:4N} = D\bar{\lambda}_r + \bar{\lambda}_u \left(-\frac{\bar{v}^2}{\bar{r}^2} + \frac{2\mu}{\bar{r}^3}\right) + \bar{\lambda}_v \left(\frac{\bar{u}\bar{v}}{\bar{r}^2}\right) \tag{43}$$

$$R_{4N+1:5N} = D\bar{\lambda}_u + \bar{\lambda}_r - \bar{\lambda}_v \frac{\bar{v}}{\bar{r}} \tag{44}$$

$$R_{5N+1:6N} = D\bar{\lambda}_v + \bar{\lambda}_u \left(\frac{2\bar{v}}{\bar{r}}\right) - \bar{\lambda}_v \left(\frac{\bar{u}}{\bar{r}}\right) \tag{45}$$

Now collocating the boundary conditions using Equations (34) – (39)

$$R_1 = \bar{r}(1) - r(t_0) \tag{46}$$

$$R_{N+1} = \bar{u}(1) - u(t_0) \tag{47}$$

$$R_{2N} = \bar{u}(N) - u(t_f) \tag{48}$$

$$R_{2N+1} = \bar{v}(1) - v(t_0) \tag{49}$$

$$R_{3N} = \bar{v}(N) - v(t_f) \tag{50}$$

$$R_{4N} = \bar{\lambda}_r(N) - \lambda_r(t_f) \tag{51}$$

Now that the six differential equations and six boundary conditions are defined, the implementation and execution of the CRBF collocation method can ensue. The key variables of interest are the state and costate development throughout the time domain. These variables are measured in MATLAB through a series of simulations at each time discretization.

The data is analyzed by evaluating the performance of the optimal trajectory and ensuring the system remained in its defined constraints and satisfied the ask of the mission: maximizing its orbit transfer distance. The accuracy and consistency of the results are compared with the shooting method.

Thrust Control Orbit Transfer

Similar to the maximum radius orbit transfer problem as previously discussed, this problem deals with optimizing the thrust elements to reach the desired prescribed final state. This thrust control orbit transfer utilizes the same parameters as described in Table 2, but the final radial distance is known as about 150% of the initial radial distance. This problem excludes the fuel consumption of the spacecraft, and implements a fixed time of 193 days. The dynamics of the problem are written as

$$\dot{r} = u \tag{52}$$

$$\dot{u} = \frac{v^2}{r} - \frac{\mu}{r^2} + T_u \tag{53}$$

$$\dot{v} = -\frac{uv}{r} + T_v \tag{54}$$

Now considering the change of the cost function, the Hamiltonian is written as

$$H = \frac{1}{2}(T_u^2 + T_v^2) + \lambda_r u + \lambda_u\left(\frac{v^2}{r} - \frac{\mu}{r^2} + T_u\right) + \lambda_v\left(-\frac{uv}{r} + T_v\right) \tag{55}$$

The costate derivatives are the same as Equations (30) – (32), since the Hamiltonian did not change with respect to the states. The control optimality condition is defined as

$$0 = \frac{\partial H}{\partial T_u} = T_u + \lambda_u \rightarrow T_u = -\lambda_u \tag{56}$$

$$0 = \frac{\partial H}{\partial T_v} = T_v + \lambda_v \rightarrow T_v = -\lambda_v \tag{57}$$

Equations (53) and (54) are rewritten as

$$\dot{u} = \frac{v^2}{r} - \frac{\mu}{r^2} - \lambda_u \tag{58}$$

$$\dot{v} = -\frac{uv}{r} - \lambda_v \tag{59}$$

The boundary conditions used for this application are the following

$$r(t_0) = 4.9081e^{11}\ ft \tag{60}$$

$$u(t_0) = 0 \tag{61}$$

$$v(t_0) = \sqrt{\frac{\mu}{r_0}} \tag{62}$$

$$r_f(t_f) = 1.5 \tag{63}$$

$$u(t_f) = 0 \tag{64}$$

$$v(t_f) = \sqrt{\frac{\mu}{r_f}} \tag{65}$$

Note Equation (63) was taken from the results of the maximum radius orbit application. The residual equations are very similar to the maximum radius orbit problem with the only difference being the exclusion of the fuel consumption of the spacecraft.

$$R_{1:N} = D\bar{r} - \bar{u} \qquad (66)$$

$$R_{N+1:2N} = D\bar{u} - \frac{\bar{v}^2}{\bar{r}} + \frac{\mu}{\bar{r}^2} - T_u \qquad (67)$$

$$R_{2N+1:3N} = D\bar{v} + \frac{\bar{u}\bar{v}}{\bar{r}} - T_v \qquad (68)$$

$$R_{3N+1:4N} = D\bar{\lambda}_r + \bar{\lambda}_u\left(-\frac{\bar{v}^2}{\bar{r}^2} + \frac{2\mu}{\bar{r}^3}\right) + \bar{\lambda}_v\left(\frac{\bar{u}\bar{v}}{\bar{r}^2}\right) \qquad (69)$$

$$R_{4N+1:5N} = D\bar{\lambda}_u + \bar{\lambda}_r - \bar{\lambda}_v\frac{\bar{v}}{\bar{r}} \qquad (70)$$

$$R_{5N+1:6N} = D\bar{\lambda}_v + \bar{\lambda}_u\left(\frac{2\bar{v}}{\bar{r}}\right) - \bar{\lambda}_v\left(\frac{\bar{u}}{\bar{r}}\right) \qquad (71)$$

Now collocating the boundary conditions using Equations (60) – (65)

$$R_1 = \bar{r}(1) - r(t_0) \qquad (72)$$

$$R_N = \bar{r}(N) - r(t_f) \qquad (73)$$

$$R_{N+1} = \bar{u}(1) - u(t_0) \qquad (74)$$

$$R_{2N} = \bar{u}(N) - u(t_f) \qquad (75)$$

$$R_{2N+1} = \bar{v}(1) - v(t_0) \qquad (76)$$

$$R_{3N} = \bar{v}(N) - v(t_f) \qquad (77)$$

The same process is used as discussed previously, and the results are created. This example uses different dynamics, constraints, and objective cost function and still produces accurate results that can be compared to the shooting method.

CHAPTER FOUR: RESULTS

This chapter presents the findings obtained from the analysis of the data collected during the study. The objective of this section is to provide a full overview of the results and insights derived from the research. The CRBF collocation method is successfully developed and executed in solving orbit trajectory problems. This establishes the CRBF collocation method in the optimal trajectory design field that can be expanded across various problems. After post processing the data, the maximum radius and thrust control are found to be consistent and accurate across the shooting method and CRBF collocation method.

Throughout this chapter, the data is analyzed in both a qualitative and quantitative manner to ensure there is a comprehensive understanding of the research problem. The presentation of the findings is supported by visual representations, including tables and figures to facilitate a clear and concise communication of the results. The results will also be discussed in the aspect of the implications it will have on the aerospace engineering field. This provides a deeper understanding of the data and allow for transparency to the relation of the existing methodologies in the field. This research contributes to the existing body of knowledge and gives further insight into solving optimal trajectory design problems. The findings are shown to have important implications on the Department of Defense (DoD), commercial, and everyday engineering applications in designing trajectories of spacecraft.

The shooting method has been one of the benchmark numerical methods in solving the optimal control problem. It involves transforming the problem into a set of IVPs. An initial guess is first used for the states and costates. After integrating the system dynamics, the shooting method compares the results with the boundary conditions and adjusts the initial guesses on its next iteration until the boundary conditions are met. This process is shown in Figure 5. Later in this chapter, the shooting method is utilized to validate the solution of the CRBF collocation method.

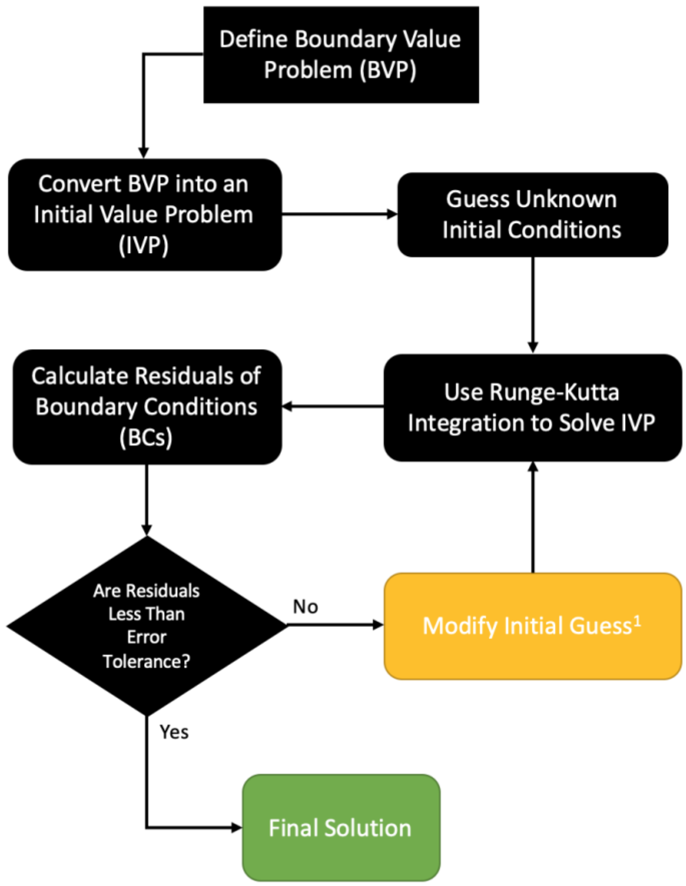

Figure 5 Shooting Method Flow Diagram

The shooting method is chosen as a validation tool for its wide adoption in optimal control problems as it is considered a baseline and standard approach for solving such problems. Note in the flow diagram different methods can be used to modify the initial guess. These methods can include Newton's method, the bisection method, the secant method, and many more. The choice of which method to use depends on the characteristics of the problem as well as what computational resources are available. For these problems, Newton's method was utilized to adjust the initial guess during the shooting method.

Maximum Radius Orbit Transfer Problem Solution

The objective of the maximum orbit radius transfer problem is to achieve the farthest final circular orbit from the initial orbit while being defined within the constraints. The CRBF collocation method is created by first defining the system parameters as shown in Table 2, then approximating the states and costates. Furthermore, the dynamics are defined along with the necessary conditions. Figure 6 shows the trajectory of the spacecraft solved with both the CRBF and the shooting methods.

Table 2 System Parameters for Maximum Radius Orbit Problem

Parameter	Value
$Thrust\ Constant, T$	0.85
$Spacecraft\ Weight, \frac{m}{g}$	$10{,}000\ lbs$
$Fuel\ Consumption, \dot{m}$	$12.9\ lb/days$
$Time\ of\ Flight, t_f$	$193\ days$
$Initial\ Radial\ Distance, r_0$	$4.9081e^{11}\ ft$

Parameter	Value
$Initial\ Radial\ Component\ of\ Velocity, u(0)$	$0\ ft/days$
$Gravitational\ Parameter, \mu$	$3.5031e^{31}\ ft^3/days^2$
$Initial\ Tangential\ Component\ of\ Velocity, v(0)$	$\sqrt{\mu/r_0}\ ft/days$
$Final\ Radial\ Distance, r_f$	$free\ state$
$Final\ Radial\ Component\ of\ Velocity, u(t_f)$	$0\ ft/days$
$Final\ Tangential\ Component\ of\ Velocity, v(t_f)$	$\sqrt{\mu/r_f}\ ft/days$

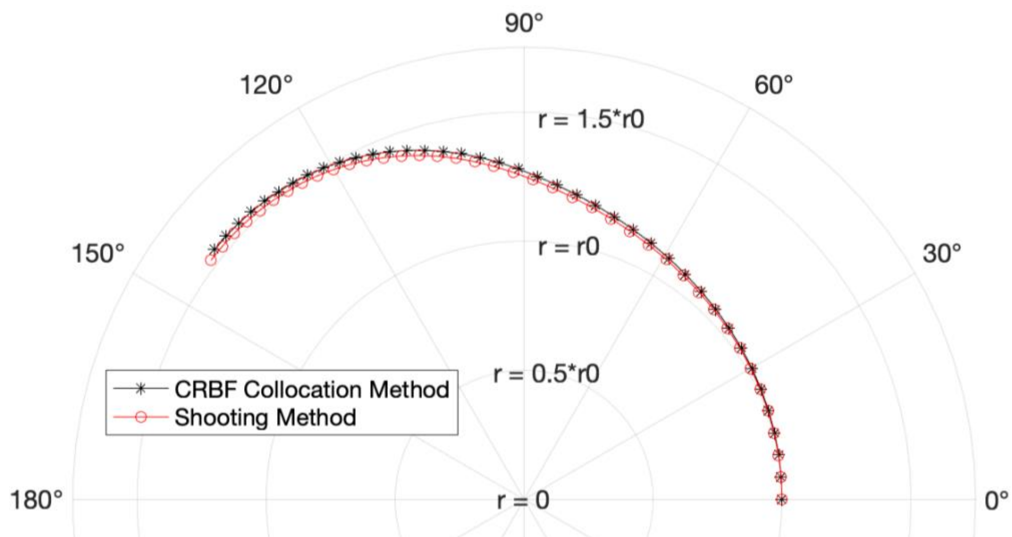

Figure 6 Orbit Trajectory Solution for Maximum Orbit Radius

Both methods do not align point to point but represent a similar trajectory to meet the objective function of maximizing the orbit radius. As shown in Table 3, the results between the two methods and optimality conditions later discussed favor the CRBF method.

Table 3 Maximum Orbit Radius Problem Results Summary

Method	Cost - Final Radial Distance (feet)
CRBF	7.5614e+11
Shooting	7.4902e+11

The CRBF produced a greater final radius, hence better achieving the objective as compared to the shooting method. The CRBF was able to output a larger final radius while staying within the constraints and system parameters of the problem. Both methods utilized fsolve as the NAE solver in MATLAB, as well as the same necessary conditions as shown in Table 4. The states and controls are comparable as well, but do not align. This non-alignment is not an issue and can be due to the nature of the methods. Both methods converged using the same dynamics and constraints. See the comparisons between the states in Figure 7.

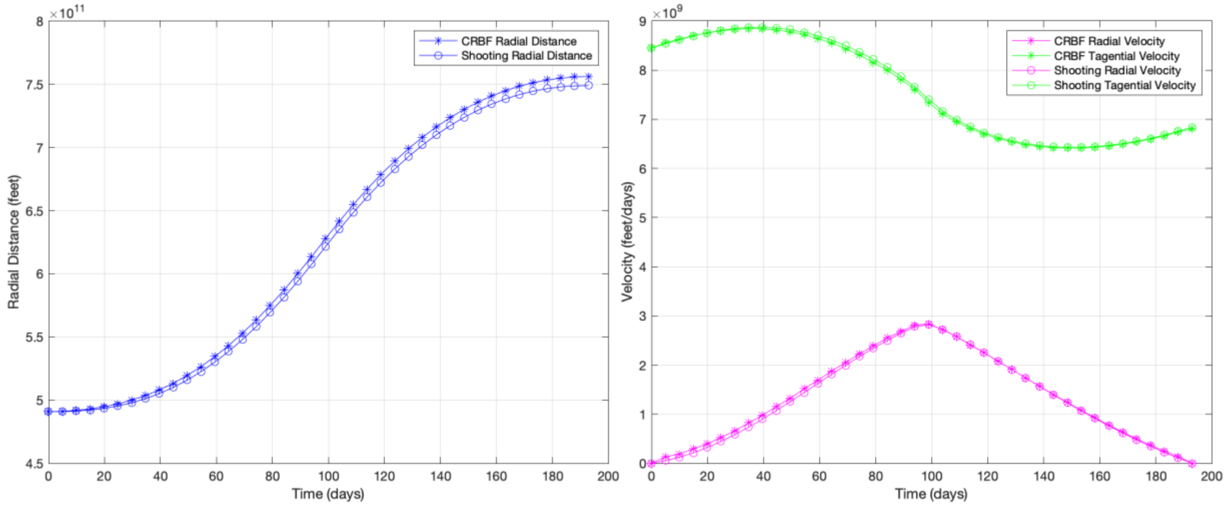

Figure 7 State Comparison for Maximum Orbit Radius Problem

As already discussed, the radial distance is comparable but does not align exactly. The CRBF method shows a high radial distance throughout the whole time domain beginning with the second time iteration (first time iteration is r_0 for both methods). Again, both radial and tagential velocities are comparable and follow the same trajectory throughout the mission. The costate comparison can be found in Figure 8.

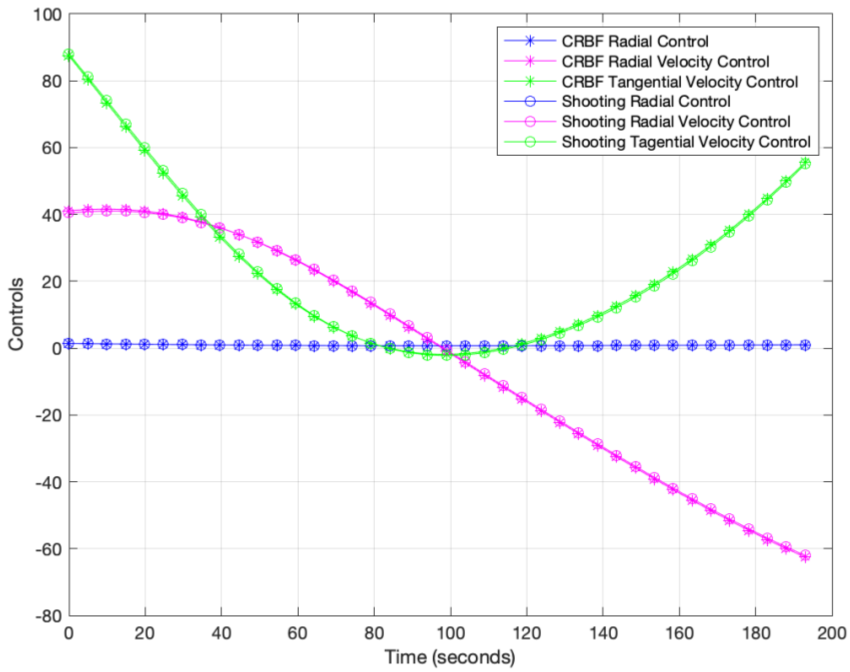

Figure 8 Costate Comparison for Maximum Orbit Raising Problem

The costates align more between the methods as compared to the states. Since the costates represent the sensitivity of the cost function to the variations of the state variables, this shows (1) the validity of the methods against the optimal behavior of the system, (2) the methods converged to a consistent solution which indicates reliability and robustness, and (3) verification of the correctness of the mathematical modeling. Table 4 describes the boundary conditions used for each method and their values.

Table 4 Maximum Orbit Radius Boundary Condition Values

Method	Boundary Conditions	Value
	$\bar{r}(1) - r(t_0)$	0
CRBF	$\bar{u}(1)$	0
	$\bar{v}(1) - v(t_0)$	0

Method	Boundary Conditions	Value
	$\bar{\lambda}_r(N) - 1$	0
	$\bar{u}(N)$	0
	$\bar{v}(N) - v(t_f)$	0
Shooting	$X(end,1) - r_f$	0
	$X(end,2)$	0
	$X(end,3) - v_f$	0
	$X(end,4) - 1$	$-2.1094e^{-15}$
	$X(end,2)$	$2.8253e^{-05}$
	$X(end,3) - v_f$	$9.5367e^{-06}$

The CRBF method proves to produce an optimal solution. All necessary conditions are very small for the CRBF method which indicate the optimality of the method and solution as shown in Table 5.

Table 5 Maximum Orbit Radius CRBF Necessary Condition Values

Method	Necessary Conditions	Value
CRBF	$R_{1:N} = D\bar{r} - \bar{u}$	$4.0634e^{-07}$
	$R_{N+1:2N} = D\bar{u} - \frac{\bar{v}^2}{\bar{r}} + \frac{\mu}{\bar{r}^2} - \frac{Tsin\phi}{m_0 - \lvert\dot{m}\rvert t}$	$-9.9995e^{-09}$
	$R_{2N+1:3N} = D\bar{v} + \frac{\bar{u}\bar{v}}{\bar{r}} - \frac{Tcos\phi}{m_0 - \lvert\dot{m}\rvert t}$	$-6.9114e^{-09}$
	$R_{3N+1:4N} = D\bar{\lambda}_r + \bar{\lambda}_u\left(-\frac{\bar{v}^2}{\bar{r}^2} + \frac{2\mu}{\bar{r}^3}\right) + \bar{\lambda}_v\left(\frac{\bar{u}\bar{v}}{\bar{r}^2}\right)$	$2.2828e^{-06}$
	$R_{4N+1:5N} = D\bar{\lambda}_u + \bar{\lambda}_r - \bar{\lambda}_v\frac{\bar{v}}{\bar{r}}$	$1.4913e^{-18}$
	$R_{5N+1:6N} = D\bar{\lambda}_v + \bar{\lambda}_u\left(\frac{2\bar{v}}{\bar{r}}\right) - \bar{\lambda}_v\left(\frac{\bar{u}}{\bar{r}}\right)$	$4.5782e^{-17}$

Thrust Control Orbit Transfer Problem Solution

The solution consists of seven parameters, all of which were defined in the structure 'solution'. This solution has 50 collocation points which is an arbitrary selection. After defining the system parameters as shown in Table 6, as well as the CRBFs, the dynamics are approximated using the CRBFs and solved using fmincon in MATLAB. The main trajectory plot of the spacecraft is shown in parallel with the solution from the shooting method in Figure 9. Note the radial distance is normalized to show the comparison between the initial and final radial distances.

Table 6 System Parameters for Thrust Control Orbit Problem

Parameter	Value
$Thrust\ Constant, T$	0.85
$Spacecraft\ Weight, \frac{m}{g}$	$10{,}000\ lbs$
$Fuel\ Consumption, \dot{m}$	$12.9\ lb/days$
$Time\ of\ Flight, t_f$	$193\ days$
$Initial\ Radial\ Distance, r_0$	$4.9081e^{11}\ ft$
$Initial\ Radial\ Component\ of\ Velocity, u(0)$	$0\ ft/days$
$Gravitational\ Parameter, \mu$	$3.5031e^{31}\ ft^3/days^2$
$Initial\ Tangential\ Component\ of\ Velocity, v(0)$	$\sqrt{\mu/r_0}\ ft/days$
$Final\ Radial\ Distance, r_f$	$7.3622e^{11}\ ft$

Parameter	Value
$Final\ Radial\ Component\ of\ Velocity, u(t_f)$	$0\ ft/days$
$Final\ Tangential\ Component\ of\ Velocity, v(t_f)$	$\sqrt{\mu/r_f}\ ft/days$

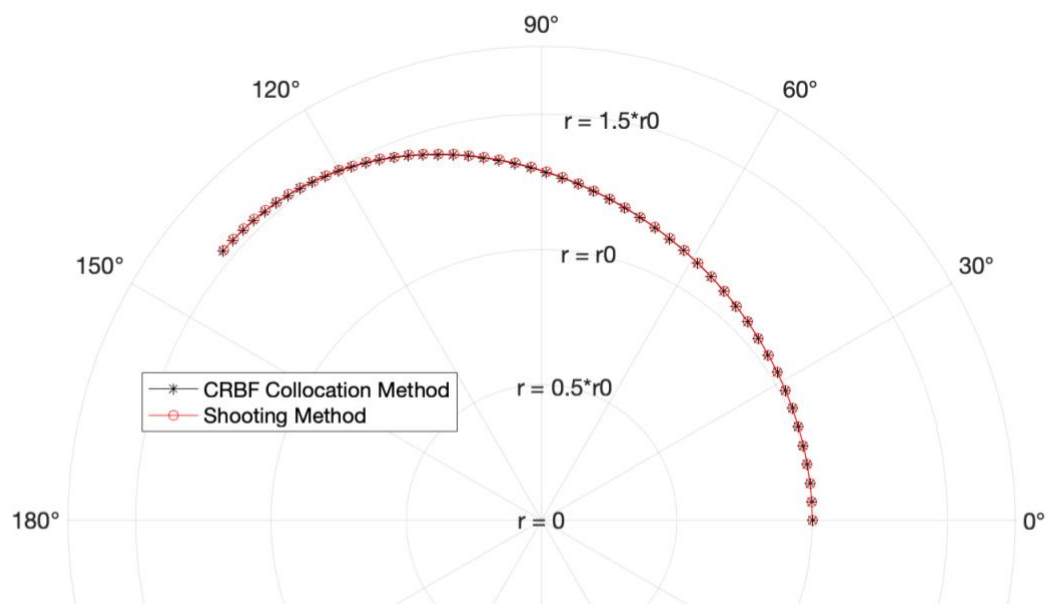

Figure 9 Orbit Trajectory Solution for Thrust Control

For each collocation point in the CRBF method, the shooting method matches the position in relation to the radial and polar angle of the result across its time discretization. The summary of the results of each method can be found in Table 7.

Table 7 Thrust Control Problem Results Summary

Method	Final Radial Distance (feet)	Cost
CRBF	7.561e+11	4.058e+16

Method	Final Radial Distance (feet)	Cost
Shooting	7.561e+11	4.054e+16

The radial distance for the final time is exactly the final state defined in the problem. Both methods are able to solve the system of dynamics while also staying within the constraints. Notice in Table 7 the CRBF method is showing a slightly higher cost function. While the CRBF collocation method implemented fmincon as its NAE solver in MATLAB, the shooting method used fsolve. Another note is that the shooting method uses guesses for the costate necessary conditions. In order to help the shooting method to converge, this had to be implemented. The CRBF method does need to use necessary condition costate guesses, as the boundary conditions for the states are defined. The comparison between the states are shown in Figure 10.

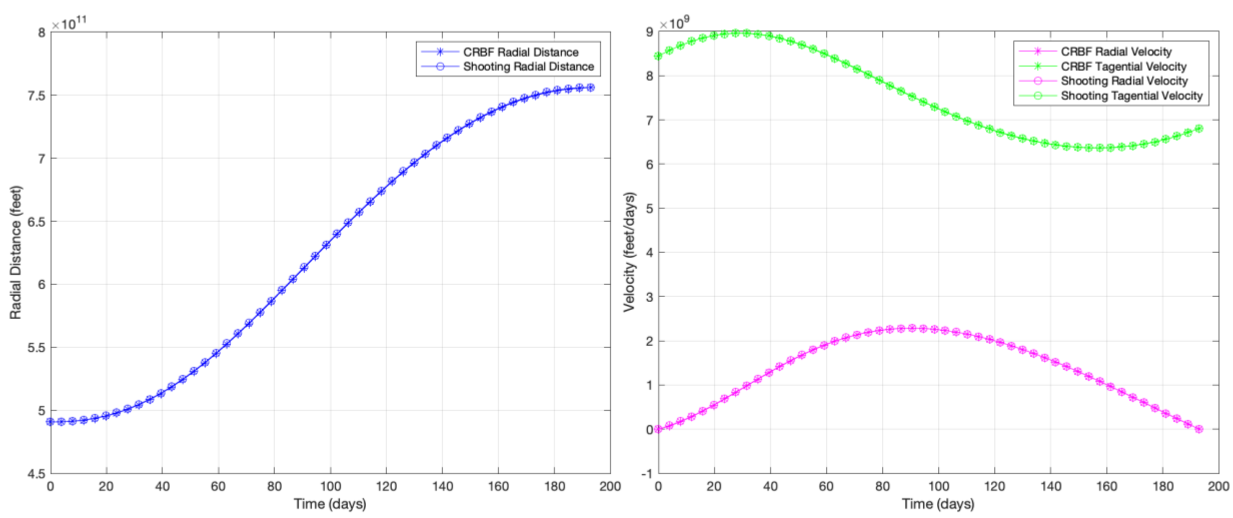

Figure 10 State Comparison for Thrust Control Problem

The states, throughout the whole time domain, align for each collocation point. The thrust control throughout the mission is shown in Figure 11. Both the CRBF and shooting methods followed the same thrust control in the radial and tangential direction. This shows the accuracy of the CRBF method, since its control follows the same path as the shooting method.

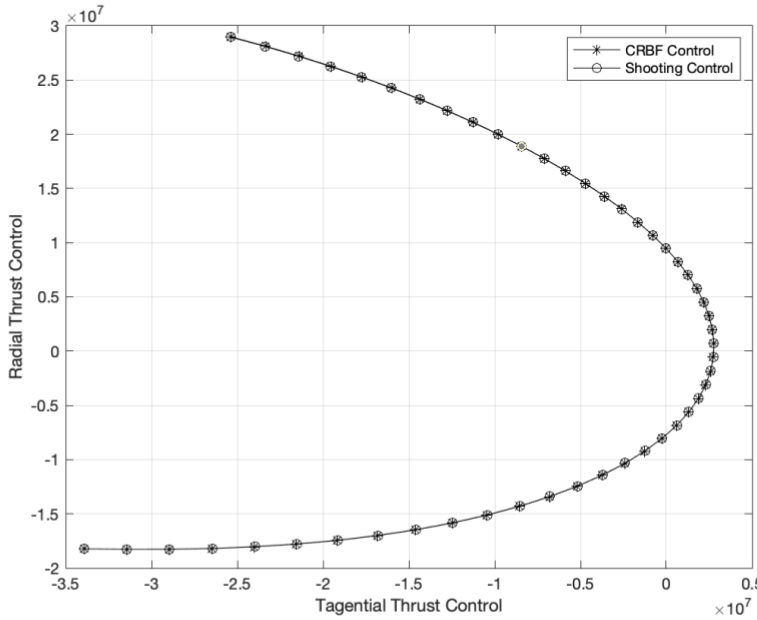

Figure 11 Thrust Control Results

The boundary conditions are shown in Table 8. For each method, the boundary conditions are set-up differently in order to allow for convergence between the two methods.

Table 8 Thrust Control Boundary Condition Values

Method	Boundary Conditions	Value
	$\bar{r}(1) - r(t_0)$	0
CRBF	$\bar{u}(1)$	0
	$\bar{v}(1) - v(t_0)$	$-2.3803e^{-18}$

Method	Boundary Conditions	Value
	$\bar{r}(N) - r(t_f)$	$7.8503e^{-19}$
	$\bar{u}(N)$	0
	$\bar{v}(N) - v(t_f)$	0
Shooting	$X(end, 1) - r_f$	$1.2207e^{-04}$
	$X(end, 2)$	$1.5795e^{-06}$
	$X(end, 3) - v_f$	$1.9073e^{-06}$
	$X(end, 4) - \nu_1$	$-5.8208e^{-11}$
	$X(end, 5) - \nu_2$	0
	$X(end, 6) - \nu_3$	$-3.3527e^{-08}$

The CRBF method solution meets the optimality conditions. Table 9 shows the values of the necessary conditions of the CRBF method which provide the necessary assessment of the optimality of the method. The closer to zero the condition is, the more optimal the solution is.

Table 9 Thrust Control CRBF Necessary Condition Values

Method	Necessary Conditions	Value
CRBF	$R_{1:N} = D\bar{r} - \bar{u}$	$-5.5511e^{-17}$
	$R_{N+1:2N} = D\bar{u} - \frac{\bar{v}^2}{\bar{r}} + \frac{\mu}{\bar{r}^2} - T_u$	$1.1999e^{-17}$
	$R_{2N+1:3N} = D\bar{v} + \frac{\bar{u}\bar{v}}{\bar{r}} - T_v$	$-1.0087e^{-16}$
	$R_{3N+1:4N} = D\bar{\lambda}_r + \bar{\lambda}_u\left(-\frac{\bar{v}^2}{\bar{r}^2} + \frac{2\mu}{\bar{r}^3}\right) + \bar{\lambda}_v\left(\frac{\bar{u}\bar{v}}{\bar{r}^2}\right)$	$3.5885e^{-17}$
	$R_{4N+1:5N} = D\bar{\lambda}_u + \bar{\lambda}_r - \bar{\lambda}_v\frac{\bar{v}}{\bar{r}}$	$6.8637e^{-17}$
	$R_{5N+1:6N} = D\bar{\lambda}_v + \bar{\lambda}_u\left(\frac{2\bar{v}}{\bar{r}}\right) - \bar{\lambda}_v\left(\frac{\bar{u}}{\bar{r}}\right)$	$-3.0104e^{-17}$

CHAPTER FIVE: CONCLUSION

This thesis aimed to create a reliable method for solving optimal spacecraft trajectory problems. After reviewing literature on previous methods, running simulations on trajectory optimization problems, and then analyzing the results, a new method has emerged in optimal trajectory design. The new CRBF collocation method is shown to solve spacecraft optimal trajectory problems. The shooting method was implemented to validate the CRBF results and verify the accuracy of the CRBF method in optimal trajectory design.

Though many methods can be utilized to solve optimal control problems, CRBFs bring classical RBF benefits such as adaptability to complex functions, being able to effectively interpolate data points, utilize both global and local collocation, employ efficient computation, and generate robust solutions while avoiding RBF sensitivity to the shape parameter values.

In this work, the CRBF collocation method has been studied for optimal spacecraft trajectory design. The indirect formulation is adopted and the necessary conditions of optimality are derived. CRBF collocation is then used to discretize the resulting two-point value problem. Then the set of NAEs are solved via a standard solver. The results show that CRBFs are able produce an optimal solution for spacecraft optimal trajectory problems. The solutions are also validated against the shooting method and showed agreement in accuracy and computational time. The CRBFs are less sensitive to the shape parameter and don’t require a need for an accurate initial guess.

While this research introduces the CRBF method in optimal trajectory design for the maximum orbit radius and thrust control problems, there are still plenty of areas CRBFs can be implemented. Further studies can look into the minimum time problem in optimal orbit design as well as three-dimensional dynamics and constraints. These problems can be simply solved using the trivial CRBF collocation method developed for optimal trajectory problems. As neural networks develop to be more intuitive and automated, CRBF collocation may be an area of study for advancement in space mission designs. While artificial intelligence and machine learning continue to advance in the aerospace engineering discipline across all necessary stakeholders when designing missions involving trajectory design, CRBFs can be a great resource when it comes to defining approximations for the system functions.

APPENDIX: SCRIPT OVERVIEW

CRBF Collocation Script

Output: Plot [solution]

collpts ()

- # of Nodes
- # of Equations
- Final Time
- RBF Type

sysparam(collpts)

- System Parameters
- Shape Parameter
- Boundary Conditions
- Initial Guess

getApprox(collpts,sysparam)

- Differential Operator

getSolution(collpts,sysparam,approx)

- Dynamics Definition
- Necessary Conditions
- NAE Solver (fmincon)

getSolError(T,solution,sysparam)

- State and Costate Error

plotSolution(T,solution,collpts,approx,solutionerror)

- Plot Solution

Shooting Method Script

Output: Plot [solution]

- **Define system parameters**
- **Implement initial guesses**
- **Utilize fsolve to solve the dynamics within the constraints**
- **Take solution from fsolve and use as initial guess into ode45 to output the solution**
- **Plot solution**

LIST OF REFERENCES

Betts, J. (1998). Survey of numerical methods for trajectory optimization. *Journal of Guidance, Control, and Dynamics 21(2)*, 193-207. https://doi.org/10.2514/2.4231

Bhatia, G. S., & Arora, G. (2016). Radial Basis Function Methods for Solving Partial Differential Equations-A Review. *Indian Journal of Science and Technology 9(45)*, 1-18. https://doi.org/10.17485/ijst/2016/v9i45/105079

Bianco, N., Bertolazzi, E., Biral, F., & Massaro, M. (2018). Comparison of direct and indirect methods for minimum lap time optimal control problems. *Vehicle System Dynamics 57(1)*, 665-696. https://doi.org/10.1080/00423114.2018.1480048

Bryson, A. E., & Ho, Y.-C. (1975). *Applied Optimal Control: Optimization, Estimation and Control* (1st ed.).

Carr, J. C., Fright, W. R., & Beatson, R. K. (1997). Surface interpolation with radial basis functions for medical imaging. *IEEE Transactions on Medical Imaging 16(1)*, 96-107. https://doi.org/10.1109/42.552059

Chen, C., Noorizadegan, A., Young, D., & Chen, C. S. (2023). On the selection of a better radial basis function and its shape parameter in interpolation problems. *Applied Mathematics and Computation 442*. 127713. https://doi.org/10.1016/j.amc.2022.127713

Cots, O., Gergaud, J., & Goubinat, D. (2018). Direct and indirect methods in optimal control with state constraints and the climbing trajectory of an aircraft. *Optimal Control Applications and Methods 39(1)*, 281-301. https://doi.org/10.1002/oca.2347

Elgohary, T., Dong, L., Junkins, J., & Atluri, S. (2014a). A simple, fast, and accurate time-integrator for strongly nonlinear dynamical systems. *CMES: Computer Modeling in Engineering & Sciences 100(3)*, 249-275. https://doi.org/10.3970/cmes.2014.100.249

Elgohary, T., Dong, L., Junkins, J., & Atluri, S. (2014b). Time domain inverse problems in nonlinear systems using collocation & radial basis functions. *CMES: Computer Modeling in Engineering & Sciences 100(1)*, 59-84. https://doi.org/10.3970/cmes.2014.100.059

Elgohary, T., Junkins, J., & Atluri, S. (2015). An RBF-collocation algorithm for orbit propagation. *Advances in Astronautical Sciences: AAS/AIAA Space Flight Mechanics Meeting*, 1021-1033.

Elnagar, G., Kazemi, M. A., & Razzaghi, M. (1995). Pseudospectral Legendre method for discretizing optimal control problems. *IEEE Transactions on Automatic Control 40(10)*, 1793-1796. https://doi.org/10.1109/9.467672

Huntington, G. T., & Rao, A. V. (2012). Comparison of global and local collocation methods for optimal control. *Journal of Guidance, Control, And Dynamics 31(2)*. https://doi.org/10.2514/1.30915

Kansa, E. J. (1990). Multiquadrics—A scattered data approximation scheme with applications to computational fluid-dynamics—II solutions to parabolic, hyperbolic and elliptic partial differential equations. *Computers & Mathematics with Applications 97(8-9)*, 147-161. https://doi.org/10.1016/0898-1221(90)90271-K

Kazem, S., Rad, J. A., & Parand, K. (2012). Radial basis functions methods for solving Fokker–Planck equation. *Engineering Analysis with Boundary Elements 36*, 181-189. https://doi.org/10.1016/j.enganabound.2011.06.012

Kelly, B., Seleit, A., & Elgohary, T. (2022). Radial optimal control software (ROCS): an adaptive solver based on radial basis functions. *AAS/AIAA Astrodynamics Specialist Conference*

Koupaei, J. A., Firouznia, M., & Hosseini S. M. M. (2018). Finding a good shape parameter of RBF to solve PDEs based on the particle swarm optimization algorithm. *Alexandria Engineering Journal 57(4)*, 3641-3652. https://doi.org/10.1016/j.aej.2017.11.024

Mirinejad, H., & Inanc, T. (2017). An RBF collocation method for solving optimal control problems. *Robotics and Autonomous Systems 87*, 219-225. https://doi.org/10.1016/j.robot.2016.10.015

Mongillo, M. (2011). Choosing Basis Functions and Shape Parameters for Radial Basis Function Methods. *SIAM Undergraduate Research Online 4*, 190-209. https://doi.org/10.1137/11S010840

Narayanaswamy, S., & Damaren, C.J. (2020). Comparison of the Legendre–Gauss pseudospectral and Hermite–Legendre–Gauss–Lobatto methods for low-thrust spacecraft trajectory optimization. *Aerospace Systems 3*, 53–70. https://doi.org/10.1007/s42401-019-00042-w

Osborne, M. R. (1969). On shooting methods for boundary value problems. *Journal of Mathematical Analysis and Applications, 27(2)*, 417-433. https://doi.org/10.1016/0022-247X(69)90059-6

Rao, A. (2010). A survey of numerical methods for optimal control. *Advances in the Astronautical Sciences*, *135(1)*, 1-32.

Seleit, A. (2022). Shape Parameter & Nodal Distribution Insensitive Radial Basis Functions for Nonlinear Optimal Control Problems. *Electronic Theses and Dissertations, 2020-, 1488*. https://stars.library.ucf.edu/etd2020/1488

Soler, M., Zou, B., & Hansen, M. (2014). Flight trajectory design in the presence of contrails: Application of a multiphase mixed-integer optimal control approach. *Transportation Research Part C: Emerging Technologies 48*. 172–194. https://doi.org/10.1016/j.trc.2014.08.009

Stryk, O. V., & Bulirsch, R. (1992). Direct and indirect methods for trajectory optimization. *Annals of Operations Research 37*, 357–373. https://doi.org/10.1007/BF02071065

Tatari, M, & Dehghan, M. (2010). A method for solving partial differential equations via radial basis functions: Application to the heat equation. *Engineering Analysis with Boundary Elements 34(3)*, 206-212. https://doi.org/10.1016/j.enganabound.2009.09.003

Tolstykh, A. I., & Shirobokov, D. A. (2003). On using radial basis functions in a “finite difference mode” with applications to elasticity problems. *Computational Mechanics 33*, 68–79. https://doi.org/10.1007/s00466-003-0501-9