

Approximate Solutions to Nonlinear Optimal Control Problems in Astrodynamics

Francesco Topputo*

Politecnico di Milano, Milano, 20156, Italy

Franco Bernelli-Zazzera†

Politecnico di Milano, Milano, 20156, Italy

Abstract

A method to solve nonlinear optimal control problems is proposed in this work. The method implements an approximating sequence of time-varying linear quadratic regulators that converge to the solution of the original, nonlinear problem. Each sub-problem is solved by manipulating the state transition matrix of the state-costate dynamics. Hard, soft, and mixed boundary conditions are handled. The presented method is a modified version of an algorithm known as “approximating sequence of Riccati equations”. Sample problems in astrodynamics are treated to show the effectiveness of the method, whose limitations are also discussed.

1 Introduction

Optimal control problems are solved with indirect or direct methods. The former stems from the calculus of variations [11, 2]; the latter uses a nonlinear programming optimization [1, 5]. Both methods require the solution of a complex set of equations (Euler–Lagrange differential equations or Karush–Kuhn–Tucker algebraic equations) for which iterative numerical methods are used. These iterative procedures implement some form of Newton’s method to find the zeros of a nonlinear function. They are initiated by providing an initial guess solution. Guessing an appropriate initial solution is not trivial, and requires a deep knowledge of the problem at hand. In indirect methods, the initial value of the Lagrange multiplier has to be provided, whose lack of physical meaning makes it difficult to formulate a good guess. In direct methods, the initial trajectory and control have to be guessed at discrete points over the whole time interval.

This paper presents an approximate method to solve nonlinear optimal control problems. This is a modification of the method known as “approximating sequence of Riccati equations”, or ASRE [3, 4]. It transforms the nonlinear dynamics and objective function into a pseudo-linear and quadratic-like structure, respectively, by using state and control dependent functions. At each iteration,

*Assistant Professor, Department of Aerospace Science and Technology, Via La Masa, 34. E-mail: francesco.topputo@polimi.it

†Full Professor, Department of Aerospace Science and Technology, Via La Masa, 34. E-mail: franco.bernelli@polimi.it

these functions are evaluated by using the solutions at the previous iteration, and therefore a series of time-varying linear quadratic regulators is treated. This sequence is solved with a state transition matrix approach, where three different final conditions are handled: final state fully specified, final state not specified, and final state not completely specified. These define hard, soft, and mixed constrained problems, respectively.

The main feature of the presented method is that it does not require guessing any initial solution or Lagrange multiplier. In fact, iterations start by evaluating the state- and control-dependent functions using the initial condition and zero control, respectively. The way the dynamics and objective function are factorized recalls the state-dependent Riccati equations (or SDRE) method [10, 13, 6]. These two methods possess some similarities, although the way they solve the optimal control problem is different. As the method is approximated, suboptimal solutions are derived. These could be used as first guess solutions for either indirect or direct methods.

2 The Nonlinear Optimal Control Problem

The optimal control problem requires that, given a set of n first-order differential equations

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t), \quad (1)$$

the m control functions $\mathbf{u}(t)$, must be determined within initial, final time t_i , t_f , such that the performance index

$$J = \varphi(\mathbf{x}(t_f), t_f) + \int_{t_i}^{t_f} L(\mathbf{x}, \mathbf{u}, t) dt, \quad (2)$$

is minimized while satisfying $n + q$ two-point conditions

$$\mathbf{x}(t_i) = \mathbf{x}_i, \quad \boldsymbol{\psi}(\mathbf{x}(t_f), t_f) = 0. \quad (3)$$

The problem consists in finding a solution that represents a stationary point of the augmented performance index

$$\bar{J} = \varphi(\mathbf{x}(t_f), t_f) + \boldsymbol{\nu}^T \boldsymbol{\psi}(\mathbf{x}(t_f), \mathbf{u}(t_f), t_f) + \int_{t_i}^{t_f} \left[L(\mathbf{x}, \mathbf{u}, t) + \boldsymbol{\lambda}^T (\mathbf{f}(\mathbf{x}, \mathbf{u}, t) - \dot{\mathbf{x}}) \right] dt, \quad (4)$$

where $\boldsymbol{\lambda}$ is the vector of costates and $\boldsymbol{\nu}$ is the multiplier of the boundary condition. The necessary conditions for optimality, also referred to as Euler–Lagrange equations, are

$$\dot{\mathbf{x}} = \frac{\partial H}{\partial \boldsymbol{\lambda}}, \quad \dot{\boldsymbol{\lambda}} = -\frac{\partial H}{\partial \mathbf{x}}, \quad \frac{\partial H}{\partial \mathbf{u}} = 0, \quad (5)$$

where H , the Hamiltonian, is

$$H(\mathbf{x}, \boldsymbol{\lambda}, \mathbf{u}, t) = L(\mathbf{x}, \mathbf{u}, t) + \boldsymbol{\lambda}^T \mathbf{f}(\mathbf{x}, \mathbf{u}, t). \quad (6)$$

The differential-algebraic system (5) must be solved together with the final boundary conditions (3) and the transversality conditions

$$\boldsymbol{\lambda}(t_f) = \left[\frac{\partial \varphi}{\partial \mathbf{x}} + \left(\frac{\partial \boldsymbol{\psi}}{\partial \mathbf{x}} \right)^T \boldsymbol{\nu} \right]_{t=t_f}, \quad (7)$$

which define a differential-algebraic parametric two-point boundary value problem whose solution supplies ν and the functions $\mathbf{x}(t)$, $\lambda(t)$, $\mathbf{u}(t)$, $t \in [t_i, t_f]$.

3 The Approximating Sequence of Riccati Equations

Let the controlled dynamics (1) be rewritten in the form

$$\dot{\mathbf{x}} = A(\mathbf{x}, t)\mathbf{x} + B(\mathbf{x}, \mathbf{u}, t)\mathbf{u} \quad (8)$$

and let the objective function (2) be rearranged as

$$J = \frac{1}{2}\mathbf{x}^T(t_f)S(\mathbf{x}(t_f), t_f)\mathbf{x}(t_f) + \frac{1}{2}\int_{t_i}^{t_f} [\mathbf{x}^T Q(\mathbf{x}, t)\mathbf{x} + \mathbf{u}^T R(\mathbf{x}, \mathbf{u}, t)\mathbf{u}] dt, \quad (9)$$

where the operators A , B , S , Q , and R have appropriate dimensions. The nonlinear dynamics (8) and the performance index (9) define an optimal control problem. The initial state, \mathbf{x}_i , is assumed given, while the final condition (ψ in (3)) can assume three different forms (see Section 4). The problem is formulated as an approximating sequence of Riccati equations. This method reduces problem (8)–(9) to a series of time-varying linear quadratic regulators that are defined by evaluating the state- and control-dependent matrices using the solution of the previous iteration (the first iteration considers the initial condition and zero control).

The initial step consists in solving *Problem 0*, defined as follows.

$$\begin{aligned} \dot{\mathbf{x}}^{(0)} &= A(\mathbf{x}_i, t)\mathbf{x}^{(0)} + B(\mathbf{x}_i, \mathbf{0}, t)\mathbf{u}^{(0)}, \\ J &= \frac{1}{2}\mathbf{x}^{(0)T}(t_f)S(\mathbf{x}_i, t_f)\mathbf{x}^{(0)}(t_f) + \frac{1}{2}\int_{t_i}^{t_f} [\mathbf{x}^{(0)T} Q(\mathbf{x}_i, t)\mathbf{x}^{(0)} + \mathbf{u}^{(0)T} R(\mathbf{x}_i, \mathbf{0}, t)\mathbf{u}^{(0)}] dt. \end{aligned}$$

Problem 0 is a standard time-varying linear quadratic regulator (TVLQR), as the arguments of A , B , S , Q , and R are all given except for the time. This problem is solved to yield $\mathbf{x}^{(0)}(t)$ and $\mathbf{u}^{(0)}(t)$, $t \in [t_i, t_f]$, where the superscript denotes the problem the solution refers to.

At a generic, subsequent iteration, *Problem k* has to be solved. This is defined as follows.

$$\begin{aligned} \dot{\mathbf{x}}^{(k)} &= A(\mathbf{x}^{(k-1)}(t), t)\mathbf{x}^{(k)} + B(\mathbf{x}^{(k-1)}(t), \mathbf{u}^{(k-1)}(t), t)\mathbf{u}^{(k)}, \\ J &= \frac{1}{2}\mathbf{x}^{(k)T}(t_f)S(\mathbf{x}^{(k-1)}(t_f), t_f)\mathbf{x}^{(k)}(t_f) \\ &\quad + \frac{1}{2}\int_{t_i}^{t_f} [\mathbf{x}^{(k)T} Q(\mathbf{x}^{(k-1)}(t), t)\mathbf{x}^{(k)} + \mathbf{u}^{(k)T} R(\mathbf{x}^{(k-1)}(t), \mathbf{u}^{(k-1)}(t), t)\mathbf{u}^{(k)}] dt. \end{aligned}$$

Problem k is again a TVLQR; note that $\mathbf{x}^{(k-1)}$ and $\mathbf{u}^{(k-1)}$ are the solutions of Problem $k-1$, achieved at previous iteration. Solving Problem k yields $\mathbf{x}^{(k)}(t)$ and $\mathbf{u}^{(k)}(t)$, $t \in [t_i, t_f]$.

Iterations continue until a certain convergence criterion is satisfied. In the present implementation of the algorithm, the convergence is reached when

$$\|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\|_\infty = \max_{t \in [t_i, t_f]} \{|x_j^{(k)}(t) - x_j^{(k-1)}(t)|, j = 1, \dots, n\} \leq \varepsilon \quad (10)$$

where ε is a prescribed tolerance. That is, iterations terminate when the difference between each component of the state, evaluated for all times, changes by less than ε between two consecutive iterations.

4 Solution of the Time-Varying Linear Quadratic Regulator by the State Transition Matrix

With the approach sketched in Section 3, a fully nonlinear optimal control problem is reduced to a sequence of time-varying linear quadratic regulators. These can be solved a number of times to achieve an approximate solution of the original, nonlinear problem. This is done by exploiting the structure of the problem as well as its state transition matrix. This scheme differs from that implemented in [3, 4], and, in part, is described in [2].

Suppose the following dynamics are given,

$$\dot{\mathbf{x}} = A(t)\mathbf{x} + B(t)\mathbf{u}, \quad (11)$$

together with the quadratic objective function

$$J = \frac{1}{2}\mathbf{x}^T(t_f)S(t_f)\mathbf{x}(t_f) + \frac{1}{2}\int_{t_i}^{t_f} [\mathbf{x}^T Q(t)\mathbf{x} + \mathbf{u}^T R(t)\mathbf{u}] dt, \quad (12)$$

where Q and R , S are positive semi-definite and positive definite time-varying matrices with appropriate dimensions, respectively. The Hamiltonian of this problem is

$$H = \frac{1}{2} [\mathbf{x}^T Q(t)\mathbf{x} + \mathbf{u}^T R(t)\mathbf{u}] + \boldsymbol{\lambda}^T [A(t)\mathbf{x} + B(t)\mathbf{u}], \quad (13)$$

and the optimality conditions (5) read

$$\dot{\mathbf{x}} = A(t)\mathbf{x} + B(t)\mathbf{u}, \quad (14)$$

$$\dot{\boldsymbol{\lambda}} = -Q(t)\mathbf{x} - A^T(t)\boldsymbol{\lambda}, \quad (15)$$

$$0 = R(t)\mathbf{u} + B^T(t)\boldsymbol{\lambda}. \quad (16)$$

From equation (16) it is possible to get

$$\mathbf{u} = -R^{-1}(t)B^T(t)\boldsymbol{\lambda}, \quad (17)$$

which can be substituted into (14)–(15) to yield

$$\dot{\mathbf{x}} = A(t)\mathbf{x} - B(t)R^{-1}(t)B^T(t)\boldsymbol{\lambda}, \quad (18)$$

$$\dot{\boldsymbol{\lambda}} = -Q(t)\mathbf{x} - A^T(t)\boldsymbol{\lambda}. \quad (19)$$

In a compact form, equations (18)–(19) can be arranged as

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\boldsymbol{\lambda}} \end{bmatrix} = \begin{bmatrix} A(t) & -B(t)R^{-1}(t)B^T(t) \\ -Q(t) & -A^T(t) \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \boldsymbol{\lambda} \end{bmatrix}. \quad (20)$$

Since (20) is a system of linear differential equations, the exact solution can be written as

$$\mathbf{x}(t) = \phi_{xx}(t_i, t)\mathbf{x}_i + \phi_{x\lambda}(t_i, t)\boldsymbol{\lambda}_i, \quad (21)$$

$$\boldsymbol{\lambda}(t) = \phi_{\lambda x}(t_i, t)\mathbf{x}_i + \phi_{\lambda\lambda}(t_i, t)\boldsymbol{\lambda}_i, \quad (22)$$

where the functions ϕ_{xx} , $\phi_{x\lambda}$, $\phi_{\lambda x}$, and $\phi_{\lambda\lambda}$ are the components of the state transition matrix, which can be found by integrating the following dynamics

$$\begin{bmatrix} \dot{\phi}_{xx} & \dot{\phi}_{x\lambda} \\ \dot{\phi}_{\lambda x} & \dot{\phi}_{\lambda\lambda} \end{bmatrix} = \begin{bmatrix} A(t) & -B(t)R^{-1}(t)B^T(t) \\ -Q(t) & -A^T(t) \end{bmatrix} \begin{bmatrix} \phi_{xx} & \phi_{x\lambda} \\ \phi_{\lambda x} & \phi_{\lambda\lambda} \end{bmatrix}, \quad (23)$$

with initial conditions

$$\phi_{xx}(t_i, t_i) = \phi_{\lambda\lambda}(t_i, t_i) = I_{n \times n}, \quad \phi_{x\lambda}(t_i, t_i) = \phi_{\lambda x}(t_i, t_i) = 0_{n \times n}. \quad (24)$$

If both \mathbf{x}_i and $\boldsymbol{\lambda}_i$ were given, it would be possible to compute $\mathbf{x}(t)$ and $\boldsymbol{\lambda}(t)$ through (21)–(22), and therefore the optimal control function $\mathbf{u}(t)$ with (17). The initial condition is assumed given, whereas the computation of $\boldsymbol{\lambda}_i$ depends on the final condition, which, in the present algorithm, can be defined in three different ways.

4.1 Hard Constrained Problem

In a hard constrained problem (HCP) the value of the final state is fully specified, $\mathbf{x}(t_f) = \mathbf{x}_f$, and therefore (12) does not account for S . The value of $\boldsymbol{\lambda}_i$ can be found by writing (21) at final time

$$\mathbf{x}_f = \phi_{xx}(t_i, t_f) \mathbf{x}_i + \phi_{x\lambda}(t_i, t_f) \boldsymbol{\lambda}_i, \quad (25)$$

and solving for $\boldsymbol{\lambda}_i$; i.e.,

$$\boldsymbol{\lambda}_i(\mathbf{x}_i, \mathbf{x}_f, t_i, t_f) = \phi_{x\lambda}^{-1}(t_i, t_f) [\mathbf{x}_f - \phi_{xx}(t_i, t_f) \mathbf{x}_i]. \quad (26)$$

4.2 Soft Constrained Problem

In a soft constrained problem (SCP) the final state is not specified, and thus S in (12) is a $n \times n$ positive definite matrix. The transversality condition (7) sets a relation between the state and costate at final time

$$\boldsymbol{\lambda}(t_f) = S(t_f) \mathbf{x}(t_f), \quad (27)$$

which can be used to find $\boldsymbol{\lambda}_i$. This is done by writing (21)–(22) at final time and using (27)

$$\mathbf{x}(t_f) = \phi_{xx}(t_i, t_f) \mathbf{x}_i + \phi_{x\lambda}(t_i, t_f) \boldsymbol{\lambda}_i, \quad (28)$$

$$S(t_f) \mathbf{x}(t_f) = \phi_{\lambda x}(t_i, t_f) \mathbf{x}_i + \phi_{\lambda\lambda}(t_i, t_f) \boldsymbol{\lambda}_i. \quad (29)$$

Equations (28)–(29) represent a linear algebraic system of $2n$ equations in the $2n$ unknowns $\{\mathbf{x}(t_f), \boldsymbol{\lambda}_i\}$. The system can be solved by substitution to yield

$$\boldsymbol{\lambda}_i(\mathbf{x}_i, t_i, t_f) = [\phi_{\lambda\lambda}(t_i, t_f) - S(t_f)\phi_{x\lambda}(t_i, t_f)]^{-1} [S(t_f)\phi_{xx}(t_i, t_f) - \phi_{\lambda x}(t_i, t_f)] \mathbf{x}_i. \quad (30)$$

4.3 Mixed Constrained Problem

In a mixed constrained problem (MCP) some components of the final state are specified and some are not. Without any loss of generality, let the state be

decomposed as $\mathbf{x} = (\mathbf{y}, \mathbf{z})$, where \mathbf{y} are the p known components at final time, $\mathbf{y}(t_f) = \mathbf{y}_f$, and \mathbf{z} are remaining $n - p$ elements. The costate is decomposed accordingly as $\boldsymbol{\lambda} = (\boldsymbol{\xi}, \boldsymbol{\eta})$. With this formalism, S in (12) is $(n - p) \times (n - p)$, and it is pre- and post-multiplied by $\mathbf{z}(t_f)$. The transversality condition (7) is $\boldsymbol{\eta}(t_f) = S(t_f)\mathbf{z}(t_f)$.

The MCP is solved by partitioning the state transition matrix in a suitable form such that, at final time, equations (21)–(22) read

$$\begin{bmatrix} \mathbf{y}(t_f) \\ \mathbf{z}(t_f) \end{bmatrix} = \begin{bmatrix} \phi_{yy} & \phi_{yz} \\ \phi_{zy} & \phi_{zz} \end{bmatrix} \begin{bmatrix} \mathbf{y}_i \\ \mathbf{z}_i \end{bmatrix} + \begin{bmatrix} \phi_{y\xi} & \phi_{y\eta} \\ \phi_{z\xi} & \phi_{z\eta} \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_i \\ \boldsymbol{\eta}_i \end{bmatrix}, \quad (31)$$

$$\begin{bmatrix} \boldsymbol{\xi}(t_f) \\ \boldsymbol{\eta}(t_f) \end{bmatrix} = \begin{bmatrix} \phi_{\xi y} & \phi_{\xi z} \\ \phi_{\eta y} & \phi_{\eta z} \end{bmatrix} \begin{bmatrix} \mathbf{y}_i \\ \mathbf{z}_i \end{bmatrix} + \begin{bmatrix} \phi_{\xi\xi} & \phi_{\xi\eta} \\ \phi_{\eta\xi} & \phi_{\eta\eta} \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_i \\ \boldsymbol{\eta}_i \end{bmatrix}, \quad (32)$$

where the dependence of the state transition matrix components on t_i, t_f is omitted for brevity. From the first row of (31) it is possible to get

$$\boldsymbol{\xi}_i = \phi_{y\xi}^{-1} [\mathbf{y}_f - \phi_{yy}\mathbf{y}_i - \phi_{yz}\mathbf{z}_i] - \phi_{y\xi}^{-1} \phi_{y\eta}\boldsymbol{\eta}_i, \quad (33)$$

which can be substituted in the second row of (31) to yield

$$\mathbf{z}(t_f) = [\phi_{zy} - \phi_{z\xi}\phi_{y\xi}^{-1}\phi_{yy}]\mathbf{y}_i + [\phi_{zz} - \phi_{z\xi}\phi_{y\xi}^{-1}\phi_{yz}]\mathbf{z}_i + \phi_{z\xi}\phi_{y\xi}^{-1}\mathbf{y}_f + [\phi_{z\eta} - \phi_{z\xi}\phi_{y\xi}^{-1}\phi_{y\eta}]\boldsymbol{\eta}_i. \quad (34)$$

Equations (33)–(34), together with the transversality condition $\boldsymbol{\eta}(t_f) = S(t_f)\mathbf{z}(t_f)$, can be substituted in the second row of (32) to compute the component of the initial costate

$$\boldsymbol{\eta}_i(\mathbf{x}_i, \mathbf{y}_f, t_i, t_f) = [\tilde{\phi}_{\eta\eta}]^{-1} \mathbf{w}(\mathbf{x}_i, \mathbf{y}_f, t_i, t_f), \quad (35)$$

where

$$\tilde{\phi}_{\eta\eta} = \phi_{\eta\eta} - \phi_{\eta\xi}\phi_{y\xi}^{-1}\phi_{y\eta} - S(\phi_{z\eta} - \phi_{z\xi}\phi_{y\xi}^{-1}\phi_{y\eta}), \quad (36)$$

and

$$\begin{aligned} \mathbf{w}(\mathbf{x}_i, \mathbf{y}_f, t_i, t_f) &= [S(\phi_{zy} - \phi_{z\xi}\phi_{y\xi}^{-1}\phi_{yy}) - \phi_{\eta y} + \phi_{\eta\xi}\phi_{y\xi}^{-1}\phi_{yy}]\mathbf{y}_i \\ &+ [S(\phi_{zz} - \phi_{z\xi}\phi_{y\xi}^{-1}\phi_{yz}) + \phi_{\eta z} + \phi_{\eta\xi}\phi_{y\xi}^{-1}\phi_{yz}]\mathbf{z}_i + [S(\phi_{z\xi}\phi_{y\xi}^{-1}) - \phi_{\eta\xi}\phi_{y\xi}^{-1}]\mathbf{y}_f. \end{aligned} \quad (37)$$

Once $\boldsymbol{\eta}_i$ is known, the remaining part of the initial costate, $\boldsymbol{\xi}_i$, is computed through (33), and therefore the full initial costate is obtained as a function of the initial condition, given final condition, initial, and final time; i.e. $\boldsymbol{\lambda}_i(\mathbf{x}_i, \mathbf{y}_f, t_i, t_f) = (\boldsymbol{\xi}_i(\mathbf{x}_i, \mathbf{y}_f, t_i, t_f), \boldsymbol{\eta}_i(\mathbf{x}_i, \mathbf{y}_f, t_i, t_f))$.

5 Numerical Examples

Two simple problems with nonlinear dynamics are considered to apply the developed algorithm. These correspond to the controlled relative spacecraft motion and to the controlled two-body dynamics for low-thrust transfers.

5.1 Low-Thrust Rendez-Vous

This problem is taken from the literature where a solution is available, for comparison's sake [9, 8]. Consider the planar, relative motion of two particles

in a central gravity field expressed in a rotating frame with normalized units: the length unit is equal to the orbital radius, the time unit is such that the orbital period is 2π , and the gravitational parameter is equal to 1. In these dynamics, the state, $\mathbf{x} = (x_1, x_2, x_3, x_4)$, represents the radial, tangential displacements (x_1, x_2) and the radial, tangential velocity deviations (x_3, x_4) , respectively. The control, $\mathbf{u} = (u_1, u_2)$, is made up by the radial and tangential accelerations, respectively.

The equations of motion are

$$\begin{aligned}\dot{x}_1 &= x_3, \\ \dot{x}_2 &= x_4, \\ \dot{x}_3 &= 2x_4 - (1 + x_1)(1/r^3 - 1) + u_1, \\ \dot{x}_4 &= -2x_3 - x_2(1/r^3 - 1) + u_2,\end{aligned}\tag{38}$$

with $r = \sqrt{(x_1 + 1)^2 + x_2^2}$. The initial condition is $\mathbf{x}_i = (0.2, 0.2, 0.1, 0.1)$. Two different problems are solved to test the algorithm in both hard and soft constrained conditions.

Hard Constrained Rendez-Vous. The HCP consists in minimizing

$$J = \frac{1}{2} \int_{t_i}^{t_f} \mathbf{u}^T \mathbf{u} dt\tag{39}$$

with the final, given condition $\mathbf{x}_f = (0, 0, 0, 0)$ and $t_i = 0, t_f = 1$.

Soft Constrained Rendez-Vous. The SCP considers the following objective function

$$J = \frac{1}{2} \mathbf{x}^T(t_f) S \mathbf{x}(t_f) + \frac{1}{2} \int_{t_i}^{t_f} \mathbf{u}^T \mathbf{u} dt\tag{40}$$

with $S = \text{diag}(25, 15, 10, 10)$, $t_i = 0$, and $t_f = 1$ (\mathbf{x}_f is free).

The differential equations (38) are factorized into the form of (8) as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ f(x_1, x_2)(1 + 1/x_1) & 0 & 0 & 2 \\ 0 & f(x_1, x_2) & -2 & 0 \end{bmatrix}}_{A(\mathbf{x})} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}}_B \begin{bmatrix} u_1 \\ u_2 \end{bmatrix},\tag{41}$$

with $f(x_1, x_2) = -1/[(x_1 + 1)^2 + x_2^2]^{3/2} + 1$. Thus, the problem is put into the pseudo-LQR form (8)–(9) by defining $A(\mathbf{x})$ and B as in (41) and by setting $Q = 0_{4 \times 4}$ and $R = I_{2 \times 2}$.

The two problems have been solved with the developed method. Table 1 reports the details of the HCP and SCP, whose solutions are shown in Figures 1 and 2, respectively. In Table 1, J is the objective function at the final iteration, “Iter” is the number of iterations, and the “CPU Time” is the computational time (this refers to an Intel Core 2 Duo 2 GHz with 4 GB RAM running Mac OS X 10.6). The termination tolerance ε in (10) is 10^{-9} . The optimal solutions found replicate those already know in literature [9, 8], so indicating the effectiveness of the developed method.

Problem	J	Iter	CPU Time (s)
HCP	0.9586	5	0.375
SCP	0.5660	6	0.426

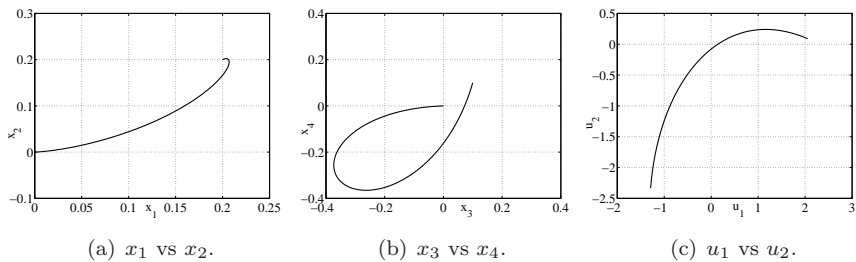


Figure 1: Hard constrained rendez-vous.

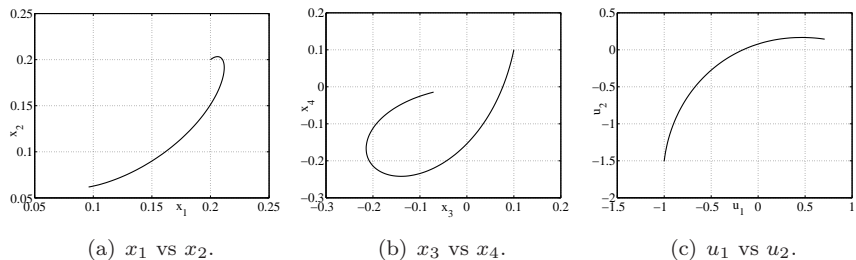


Figure 2: Soft constrained rendez-vous.

5.2 Low-thrust Orbital Transfer

In this problem the controlled, planar Keplerian motion of a spacecraft in polar coordinates is studied. The dynamics are written in scaled coordinates where the length unit corresponds to the radius of the initial orbit, the time unit is such that its period is 2π , and the gravitational parameter is 1. The state, $\mathbf{x} = (x_1, x_2, x_3, x_4)$, is made up by the radial distance from the attractor (x_1), the phase angle (x_2), the radial velocity (x_3), and the transversal velocity (x_4), whereas the control, $\mathbf{u} = (u_1, u_2)$ corresponds to the radial and transversal accelerations, respectively [12, 7]. The equations of motion are

$$\begin{aligned}
 \dot{x}_1 &= x_3, \\
 \dot{x}_2 &= x_4, \\
 \dot{x}_3 &= x_1 x_4^2 - 1/x_1^2 + u_1, \\
 \dot{x}_4 &= -2x_3 x_4/x_1 + u_2/x_1,
 \end{aligned} \tag{42}$$

and the objective function is

$$J = \frac{1}{2} \int_{t_i}^{t_f} \mathbf{u}^T \mathbf{u} dt, \tag{43}$$

with $t_i = 0$ and $t_f = \pi$. The initial state corresponds to the conditions at the initial orbit; i.e., $\mathbf{x}_i = (1, 0, 0, 1)$. Two different HCP are solved, which correspond

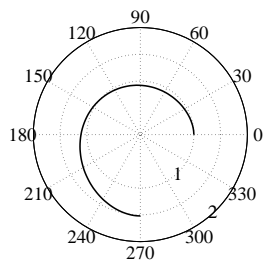
to the final states $\mathbf{x}_f = (1.52, \pi, 0, 1.52^{-3/2})$ and $\mathbf{x}_f = (1.52, 1.5\pi, 0, 1.52^{-3/2})$, respectively. (This set-up mimics an Earth–Mars low-thrust transfer.) The dynamics (42) and the objective function (43) are put in the form (8)–(9) by defining $Q = 0_{4 \times 4}$, $R = I_{2 \times 2}$, and

$$A(\mathbf{x}) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ x - 1/x_1^3 & 0 & 0 & x_1 x_4 \\ 0 & 0 & -2x_4/x_1 & 0 \end{bmatrix}, \quad B(\mathbf{x}) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1/x_1 \end{bmatrix}. \quad (44)$$

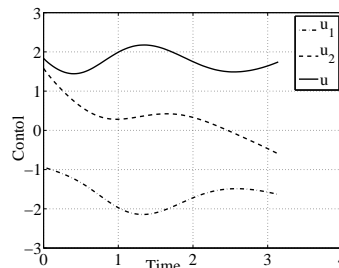
The two HPC have been solved with the developed method. The solutions details are reported in Table 2, whose columns have the same meaning as in Table 1. It can be seen that more iterations and an increased computational burden are required to solve this problem. The solution with $x_{2,f} = 1.5\pi$ is reported in Figure 3.

Table 2: Earth–Mars transfers details.

Problem	J	Iter	CPU Time (s)
$x_{2,f} = \pi$	0.5298	22	5.425
$x_{2,f} = 1.5\pi$	4.8665	123	41.831



(a) Transfer trajectory.



(b) Control profile.

Figure 3: Orbital transfer with $x_{2,f} = 1.5\pi$.

6 Conclusion

In this paper an approximated method to solve nonlinear optimal control problems has been presented, with applications to sample cases in astrodynamics. With this method the nonlinear dynamics and objective function are factorized in a pseudo-linear and quadratic-like forms, which are similar to those used in the state-dependent Riccati equation approach. Once in this form, a number of time-varying linear quadratic regulator problems are solved, with the nonlinear functions that are evaluated using the solution of the previous step. A state transition matrix approach is used to deal with the time-varying linear quadratic regulators. The results show effectiveness of the method, which can be used to either have suboptimal solutions or provide initial solutions to more accurate optimizers.

References

- [1] J.T. Betts. *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*. SIAM, 2010.
- [2] A.E. Bryson and Y.C. Ho. *Applied Optimal Control*. John Wiley & Sons, New York, 1975.
- [3] T. Cimen and S.P. Banks. Global Optimal Feedback Control for General Non Linear System with Non-Quadratic Performance Criteria. *System & Control Letters*, 53:327–346, 2004.
- [4] T. Cimen and S.P. Banks. Nonlinear Optimal Tracking Control with Application to Super-Tankers for Autopilot Design. *Automatica*, 40:1845–1863, 2004.
- [5] B. Conway. *Spacecraft Trajectory Optimization*, chapter Spacecraft Trajectory Optimization Using Direct Transcription and Nonlinear Programming, pages 37–78. Cambridge University Press, 2010.
- [6] C.P. Mracek and J.R. Cloutier. Control Designs for the Nonlinear Benchmark Problem via the State-Dependent Riccati Equation Method. *International Journal of Robust and Nonlinear Control*, 8:401–433, 1998.
- [7] A. Owis, F. Topputo, and F. Bernelli-Zazzera. Radially Accelerated Optimal Feedback Orbits in Central Gravity Field with Linear Drag. *Celestial Mechanics and Dynamical Astronomy*, 103(1):1–16, January 2009.
- [8] C. Park, V. Guibout, and D.J. Scheeres. Solving Optimal Continuous Thrust Rendezvous Problems with Generating Functions. *Journal of Guidance, Control, and Dynamics*, 29(2):321–331, March–April 2006.
- [9] C. Park and D.J. Scheeres. Solution of Optimal Feedback Control Problems with General Boundary Conditions Using Hamiltonian Dynamics and Generating Functions. *Automatica*, 42(5):869–875, May 2006.
- [10] J.D. Pearson. Approximation Methods in Optimal Control. *Journal of Electronics and Control*, 13:453–469, 1962.
- [11] L.S. Pontryagin, V.G. Boltyanskii, R.V. Gamkrelidze, and E.F. Mishchenko. *The Mathematical Theory of Optimal Processes*. John Wiley & Sons, New York, 1962.
- [12] F. Topputo, A. Owis, and F. Bernelli-Zazzera. Analytical Solution of Optimal Feedback Control for Radially Accelerated Orbits. *Journal of Guidance, Control and Dynamics*, 31(5):1352–1359, September–October 2008.
- [13] A. Wernli and G. Cook. Suboptimal Control for the Nonlinear Quadratic Regulator Problem. *Automatica*, 11:75–84, 1975.