Engineering Notes

Path Planning Using a Novel Finite Horizon Suboptimal Controller

Ali Heydari$^*$ and S. N. Balakrishnan$^+$
Missouri University of Science and Technology,
Rolla, Missouri 65409
DOI: 10.2514/1.59127

I. Introduction

FINITE horizon optimal control of nonlinear systems is a challenging problem due to the time dependency of the associated Hamilton–Jacobi–Bellman (HJB) partial differential equation. Developments in calculating closed form solution to the problem include [1] which gives a solution for prescribed initial condition and time-to-go. Using a series expansions of the solution is considered in [2,3]. The drawback of series-based methods is the limited domain of convergence; hence, they are not suitable for systems with high nonlinearities [3]. For the intelligent control methods, the interested reader may refer to [4] and the references therein.

The state-dependent Riccati equation (SDRE) method [5] for infinite horizon optimal control of nonlinear systems provides the motivation for this study; a state-dependent differential Riccati equation is introduced in this work to provide an approximate closed-form solution to the finite horizon optimal control problem. The relation between the equation and the HJB equation is investigated. Then, an approximate method is suggested for solving the differential Riccati equation. The performance of the developed technique is investigated with path-planning problems for the approach and landing (A&L) phase of a reusable launch vehicle (RLV), such that the vehicle lands in a fixed and prespecified downrange with the least possible vertical velocity and flight-path angle. Interested readers may refer to [6] for a literature review on the available methods for the A&L problem.

II. Development of the Controller

A. Problem Formulation

The nonlinear input-affine system considered in this study is assumed to be of the form

$$\dot{x}(t) = f(x(t)) + B(x(t))u(t)$$  \hspace{1cm} (1)

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ are the state and control vectors at time $t$, respectively, and $n$ and $m$ are the order of the system and number of inputs, respectively. Functions $f: \mathbb{R}^n \rightarrow \mathbb{R}^n$ and $B: \mathbb{R}^n \rightarrow \mathbb{R}^{n \times m}$ represent the continuous dynamics of the system. For simplicity in the notation, the argument $t$ of the vectors $x(t)$ and $u(t)$ is omitted in some places in this Note. Associated cost function considered in this study is given by

$$J = \frac{1}{2} x^T(t) S x(t) + \frac{1}{2} \int_{t_0}^{t_f} (x^T Q x + u^T R u) \, dt$$  \hspace{1cm} (2)

where the matrices $S$ and $Q$ are assumed to be positive semidefinite, and the matrix $R$ is a positive definite matrix. Note that the method presented here admits state-dependent $Q$ and $R$, also, i.e., $Q(x)$ and $R(x)$, respectively. The problem is finding the controller that minimizes the cost function $J$ subject to the state equation [Eq. (1)].

B. Finite State-Dependent Riccati Equation

The approach developed here for solving finite horizon optimal nonlinear problems is inspired by the state-dependent Riccati equation (SDRE) method [5] and is called the finite horizon state-dependent Riccati equation (Finite-SDRE) method. Note that in the finite horizon problems, the solution is time-dependent, and a differential equation, rather than an algebraic one used in the SDRE method for infinite horizon problem, needs to be solved to calculate the control values. Rewriting the state space equation [Eq. (1)] in a linear-like form as

$$\dot{x}(t) = A(x)x(t) + B(x)u(t)$$  \hspace{1cm} (3)

where $A(x)x(t) = f(x)$, $\forall x \in \mathbb{R}^n$, following the method of solving a differential Riccati equation (DRE) for finite horizon optimal control of linear systems, the state-dependent DRE given as follows is proposed to be solved for the approximate finite horizon optimal control

$$P(x, t) A(x) + A^T(x) P(x, t) + Q - P(x, t) B(x) R^{-1} B^T(x) P(x, t) = -\dot{P}(x, t)$$  \hspace{1cm} (4)

where $\dot{P}(x, t)$ denotes the total time derivative of matrix $P(x, t)$. The final condition is given by

$$P(x, t_f) = S$$  \hspace{1cm} (5)

and the control is then calculated in a feedback form as

$$u(x, t) = -R^{-1} B^T(x) P(x, t) x(t)$$  \hspace{1cm} (6)

$^*$Ph.D. Student, Dept. of Mechanical and Aerospace Engineering; heydari.ali@ieee.org. Student Member AIAA.

$+$Curators’ Professor of Aerospace Engineering, Dept. of Mechanical and Aerospace Engineering; bala@mst.edu. Associate Fellow AIAA.
For more information about forming the linear-like state equation [Eq. (3)] from Eq. (1), one may refer to [7]. The following assumption is made for guaranteeing a positive definite solution to the DRE [Eq. (4)] for $t \in [t_0, t_f]$.

Assumption I: Pairs $\{A(x), B(x)\}$ and $\{A(x), Q^{1/2}\}$ are pointwise stabilizable and observable for all $x \in \mathbb{R}^n$, respectively, where the Cholesky decomposition of $Q$ is denoted by $Q^{1/2}$.

C. Supporting Theory

The relation between the proposed method and the exact optimal solution to the problem is analyzed here. It is well known that the finite horizon optimal control can be obtained from solving the HJB equation given in Eq. (7):

$$-J^*(x, t) = J^*_x(x, t)f(x) + \frac{1}{2} x^T Q x$$

with the final condition

$$J^*(x, t_f) = \frac{1}{2} x^T S_x$$

where $J^*(x, t)$ denotes the optimal cost-to-go and subscripts $t$ and $x$ denote the corresponding partial derivatives of $J^*(x, t)$. Once $J^*(x, t)$ is calculated, the optimal control is given by

$$u^*(x, t) = -R^{-1}B^T(x)J^*(x, t)$$

Because $J^*(x, t)$ is a positive definite parameter on $t \in [t_0, t_f]$, it is assumed that it can be expressed as

$$J^*(x, t) = \frac{1}{2} x^T P(x, t)x$$

for some symmetric matrix-valued function $P : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n \times n}$ that is pointwise positive definite on $t \in [t_0, t_f]$. Assume the factorization of $f(x) = A(x)x$. Note that every other factorization of $f(x) = A(x)x$ can be parameterized by $A(x) = \breve{A}(x) + E(x, t)$ for some matrix-valued function $E : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n \times n}$, where $E(x, t_x) = 0, \forall x \in \mathbb{R}^n, \forall t \in [t_0, t_f]$. Using Eq. (10) in Eq. (7), as shown in detail in [8], leads to

$$\frac{1}{2} x^T \breve{P}(x, t)x = x^T \left( P(x, t)\breve{A}(x) + \frac{1}{2} Q \right) x$$

where denoting the $i$th element of the state vector $x$ by $x_i, 1 \leq i \leq n$, the $i$th element of the $j$th row of matrix $M$ by $M_{ij}$, and $P_\xi \equiv \partial \breve{P}(x, t) / \partial x_i$, the matrix $\Omega$ is defined as $\Omega \equiv 1/2 \sum_{i,j} P_{x_i x_j}B(x)R^{-1}B^T(x)P_{x_i x_j}$. In order for Eq. (11) to hold for all (the Mathematical expression: $x \in \mathbb{R}^n$) one needs

$$-\breve{P}(x, t) = P(x, t)\breve{A}(x) + \breve{A}^T(x)P(x, t) + Q - P(x, t)B(x)R^{-1}B^T(x)P(x, t) + \Omega + W(x, t) + W^T(x, t)$$

for some matrix-valued function $W : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n \times n}$ where $W(x, t_x) = 0, \forall x \in \mathbb{R}^n, \forall t \in [t_0, t_f]$. Because of full-rankness of the matrix $P(x, t) = W(x, t) / W(t)$, it is always possible to calculate $E(x, t) = \breve{A}(x) + E(x, t)$: in this case, the matrix $W(x, t)$ will have the property of $E(x, t)$, $t \in [t_0, t_f]$. From Eq. (10), it is not possible to solve the linearized Euler-Langrange equation [9], i.e., Eq. (12) is equivalent to Eq. (13) where $A(x) = \breve{A}(x) + E(x, t)$:

$$P(x, t)A(x) + A^T(x)P(x, t) + Q - P(x, t)B(x)R^{-1}B^T(x)P(x, t) + \Omega = -\breve{P}(x, t)$$

D. Approximate Solution

There is a problem in solving Eq. (4) with the final condition [Eq. (5)], that is, because the states in the future are not known ahead of time, one cannot calculate the state-dependent coefficients to integrate Eq. (4) backward from $t_f$ to $t_1$ and end with the unknown at time $t$, i.e., the matrix $P(x, t)$. To remedy this problem, an approximate analytical approach is proposed in this Note. Following the developed method in [10, 11] for a linear system to solve a constant coefficient DRE, the following procedure is proposed for the nonlinear problem at hand. Solve the Algebraic Riccati Equation (ARE) given as follows:

$$P_{ss}(x)A(x) + A^T(x)P_{ss}(x) - P_{ss}(x)B(x)R^{-1}B^T(x)P_{ss}(x) + Q = 0$$

Subtracting Eq. (15) from Eq. (4) results in

$$(P(x, t) - P_{ss}(x))A(x) + A^T(x)(P(x, t) - P_{ss}(x))$$

$$- P(x, t)B(x)R^{-1}B^T(x)P(x, t)$$

$$+ P_{ss}(x)B(x)R^{-1}B^T(x)P_{ss}(x) = -\breve{P}(x, t)$$

Using the change of variable of $K(x, t) \equiv (P(x, t) - P_{ss}(x))^{-1}$ in Eq. (16) leads to

$$\breve{K}(x, t) = A(x)K(x, t) + K(x, t)A^T(x) - B(x)R^{-1}B^T(x)$$

where $A(x) \equiv A(x) - B(x)R^{-1}B^T(x)P_{ss}(x)$. The final condition of Eq. (17) is

$$K(x, t_f) = (S - P_{ss}(x))^{-1}$$

Interestingly, Eq. (17) is no longer a nonlinear equation, and a closed-form solution can be found for $K(x, t)$. First, define $A_{sl}$ as $A_{sl}(x)$ evaluated at a current state vector, and $B$ as $B(x)$ evaluated at a current state vector, i.e.

$$A_{sl} \equiv A_{sl}(x)_{x=x(t)}; \quad B \equiv B(x)_{x=x(t)}$$

By using $A_{sl}$ and $B$ in Eq. (17) instead of $A(x)$ and $B(x)$, respectively, the solution, $K(x, t)$, can be written as

$$K(x, t) = e^{\lambda(t_f)}(K(x, t_f) - D)e^{\lambda(t_f)} + D$$

where $D$ is the solution to the algebraic Lyapunov equation

$$A_{sl}D + D A_{sl}^T = BR^{-1}B^T$$

In summary, these steps need to be done online at each time step: solve ARE [Eq. (15)] and calculate [Eq. (18)], and then [Eq. (20)] needs to be solved; that result will then be used in Eq. (19) to calculate $K(x, t)$. Having $K(x, t)$ one has

$$P(x, t) = K^{-1}(x, t) + P_{ss}(x)$$

Finally, the control can then be calculated in the feedback form as
The computational effort needed at each time step mainly includes solving an ARE, calculating a matrix exponential, and performing two matrix inversions. It is known that, at least for linear systems, if $t_0 < t_1$, the solution to the DRE in Eq. (4) converges to that of the ARE in Eq. (15), and $K(x, t)$, which is the inverse of $(P(x, t) - P_0(x))$, could become singular. To avoid this singularity, [11] suggests that the negative definite solution of ARE [Eq. (15)] be calculated instead of the positive definite solution. Note that in this case $(P(x, t) - P_0(x))$ is guaranteed to be positive definite, hence, its inverse always exists. This approach works for the nonlinear case as well. For calculation of the negative definite solution to an ARE, it suffices to flip the sign of matrix $A(x)$ and solve the ARE for the positive definite solution; then, by using the negative of $P_0(x)$, the negative definite solution to the original ARE can be obtained [11].

### III. Application: Path-Planning Problem

#### A. System Modeling

In this section, the dynamics of an RLV during landing is presented. Assuming zero cross range to the runway, the approach and landing phase of an RLV in a single vertical plane can be modeled as follows [6]:

\[
\dot{V} = -\frac{D}{m} - g \sin \gamma \tag{21}
\]

\[
\dot{Y} = \frac{L}{mV} - g \cos \gamma \tag{22}
\]

\[
\dot{h} = V \sin \gamma \tag{23}
\]

\[
\dot{X} = V \cos \gamma \tag{24}
\]

where $L = \tilde{q} S_a C_L$, $\tilde{q} = \frac{1}{2} \rho V^2$. The air density profile is assumed exponential as $\rho = \rho_0 e^{-(h/H)}$ and the lift and drag coefficients are modeled with a parabolic drag polar as $C_L = C_{L,0} + \alpha$, and $C_D = C_{D,0} + K_d C_L^2$. The Eqs. (21)–(24) represent a state space that is nonaffine in the control $\alpha$, because it contains the square of the control in the equation for $C_D$. To convert the nonaffine system into an input-affine form, the state vector is augmented with $\alpha$, and its derivative is considered as the new control [13] that is:

\[
\dot{\alpha} = \alpha \tag{25}
\]

Note that in practical situations, one is usually interested in an optimal landing in a predetermined and fixed downrange, not in a fixed time. Therefore, the independent variable of the system needs to be changed from $t$ to $X$, i.e., from the time to the downrange, for convenience. To do that, divide Eqs. (21)–(23) and Eq. (25) by Eq. (24). Denoting the derivative of a variable with respect to $X$ by the prime notation, one has

\[
V' \equiv \frac{dV}{dX} = \frac{1}{V \cos \gamma} \left( -\frac{D}{m} - g \sin \gamma \right)
\]

\[
\gamma' \equiv \frac{d\gamma}{dX} = \frac{1}{V \cos \gamma} \left( \frac{L}{mV} - g \cos \gamma \right)
\]

\[
h' \equiv \frac{dh}{dX} = \tan \gamma
\]

\[
\alpha' \equiv \frac{d\alpha}{dX} = \frac{1}{V \cos \gamma} \alpha
\]

The time becomes an incidental variable given by

\[
J = \frac{1}{2} \int_{X_0}^{X_f} \left( x^T Q x + u^T Ru \right) dX
\]

where $X_0$ and $X_f$ denote the initial downrange and the prespecified final downrange in which the RLV is supposed to land on the runway, respectively. The guidance problem based on minimizing the foregoing cost function is a finite horizon optimal control problem for a nonlinear input-affine system, and the Finite-SDRE can be used for solving this problem.

#### B. Numerical Analysis

As mentioned earlier, the factorization of the dynamics for creating the matrix $A(x)$ to be used in Finite-SDRE is not unique, and techniques suggested in [7] may be used for that purpose. The nonzero elements of the selected matrix $A(x)$ for this problem are given as follows:

\[
A_{11} = -\rho_0 \exp(-x_3/H) S_a (C_{D0} + K_d C_L^2 x_2^2) / 4m \cos x_2 - \rho_0 S_a C_{D0} / 4m \cos x_2.
\]

\[
A_{12} = -g \sin x_2 / x_1 x_2 \cos x_2.
\]

\[
A_{13} = -\rho_0 S_a C_{D0} x_1 (\exp(-x_3/H) - 1) / 4m x_2 \cos x_2.
\]

\[
A_{15} = -\rho_0 \exp(-x_3/H) S_a K_d C_L^2 x_1 x_3 / 4m \cos x_2.
\]

\[
A_{21} = -g / x_1^3.
\]

\[
A_{23} = \rho_0 \exp(-x_3/H) - 1) S_a C_{L0} / 4m x_3 \cos x_2.
\]

\[
A_{25} = \rho_0 \exp(-x_3/H) S_a C_{L0} / 4m \cos x_2 + \rho_0 S_a C_{L0} / 4m \cos x_2.
\]

\[
A_{32} = \sin(x_2) / (x_2 \cos x_2).
\]

\[
A_{41} = 1 / x_1^2 \cos x_2.
\]

**Fig. 1 State trajectories versus downrange.**
Fig. 2 Altitude and flight-path angle histories for different initial conditions and downranges.

where $A_{ij}$ denotes the $j$th element of $i$th row of $A(x)$ and $i$, $j = 1, 2, \ldots, 5$. The following values are used for the RLV parameters: $C_{Lr} = 2.3$, $C_{Dr} = 0.0975$, $K_I = 0.1819$, $J/m = 0.912$ ft$^2$/slug. The weight matrices are selected as: $R = 1$, $Q = \text{diag}(0, 0.05, 0.01, 10^{-6}, 1)$, and $S = \text{diag}(0, 10^6, 10^3, 0, 0)$. In matrix $Q$, the last element, which corresponds to the angle of attack $\alpha$, is assigned a relatively high value to force the controller to give physically admissible values. The rest of the elements of $Q$ are selected through trial and error to yield good performance. Because the goal of the problem is to land with small values of vertical velocity and flight-path angle, the second and third states, which are $\gamma$ and $h$, are penalized with high values to the corresponding diagonal elements of $S$.

The initial condition for the simulation is selected as initial height of 10,000 ft, initial velocity of 300 ft/s, and initial $\gamma$ and $\alpha$ of -30 and 10 deg, respectively. The problem is to land the RLV at a fixed downrange of 20,000 ft. Simulations were carried out with a sampling interval of 20 ft, and the resulting state trajectory histories are shown in Fig. 1. As can be seen, the controller has been able to force the flight-path angle $\gamma$ to go to zero at the touchdown. Consequently, close to zero vertical velocity can be observed at landing in the vertical velocity plot. The final value of $\gamma$ in this simulation turns out to be around -0.01 deg, and the touchdown vertical velocity is 0.06 ft/s, which is quite small. The resulting angle of attack is small enough for practical applications.

To show the performance of the proposed controller for various conditions, a few other sets were chosen, and the results are presented here. The resulting flight-path angle history and altitude history for different initial altitudes of 7000, 12,000 and 17,000 ft are shown in Fig. 2a. The same histories for the different initial flight-path angles of -50, -30, and -10 deg are given in Fig. 2b. Moreover, these results for the cases of different initial velocities of 200, 350, and 500 ft/s are plotted in Fig. 2c, and finally, the initial conditions simulated in the first simulation, but for three different horizons of 15,000, 20,000, and 25,000 ft are simulated, and the resulting histories are given in Fig. 2d. As can be seen in all of these figures, the controller has been able to design a trajectory that smoothly lands the RLV at the desired downrange, which demonstrates the potential of the method for online path planning.

IV. Conclusions

An approximate closed-loop solution to optimal control problems with finite horizon cost function is developed, and through a challenging path-planning problem, the potential of the developed method is demonstrated. Many practical control problems require a goal to be achieved in a given time, and the proposed method seems to have a lot of potential in solving such problems in closed-form.

Acknowledgment

This research was partially supported by a grant from the National Science Foundation.

References


