Received: March 13, 2024; **Accepted:** September 30, 2024.
DOI. 10.30473/coam.2024.70834.1254
Summer-Autumn (2024) Vol. 9, No. 2, (1-19)
Research Article

Open Access

Control and Optimization in Applied Mathematics - COAM

A Sub-Ordinary Approach to Achieve Near-Exact Solutions for a Class of Optimal Control Problems

Akbar Hashemi Borzabadi[1]✉, Mohammad Gholami Baladezaei[2], Morteza Ghachpazan[3]

[1]Department of Applied Mathematics, University of Science and Technology of Mazandaran, Behshahr, Iran,
[2]Department of Mathematics, Damghan Branch, Islamic Azad University, Damghan, Iran,
[3]Department of Applied Mathematics, School of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran.

✉ **Correspondence:**
Akbar Hashemi Borzabadi
E-mail:
borzabadi@mazust.ac.ir

How to Cite
Borzabadi, A.H., Gholami Baladezaei, M., Gachpazan, M. (2024). "A sub-ordinary approach to achieve near-exact solutions for a class of optimal control problems", Control and Optimization in Applied Mathematics, 9(2): 1-19.

Abstract. This paper explores the advantages of Sub-ODE strategy in deriving near-exact solutions for a class of linear and nonlinear optimal control problems (OCPs) that can be transformed into nonlinear partial differential equations (PDEs). Recognizing that converting an OCP into differential equations typically increases the complexity by adding constraints, we adopt the Sub-ODE method, as a direct method, thereby negating the need for such transformations to extract near exact solutions. A key advantage of this method is its ability to produce control and state functions that closely resemble the explicit forms of optimal control and state functions. We present results that demonstrate the efficacy of this method through several numerical examples, comparing its performance to various other approaches, thereby illustrating its capability to achieve near-exact solutions.

Keywords. Optimal control problem, Subsidiary ordinary differential equation method, Parametrization.

MSC. 49J15; 49J20; 49N05.

1 Introduction

Optimal Control Problems (OCPs) are prevalent across various scientific disciplines, including engineering and economics; see e.g., [1, 7, 15]. Consequently, finding solutions to these problems holds significant importance. Two primary strategies exist for approximating solutions: direct and indirect methods. In the indirect approach, the original problem is first transformed into a system of differential equations, which are then solved using analytical or numerical methods. Conversely, in direct methods, the problem is reformulated into a nonlinear optimization problem through techniques such as discretization and parametrization, enabling the derivation of approximate solutions for the initial OCP; refer to [2, 11, 16, 19]. It is evident that, regardless of the method employed—direct or indirect—the application of effective approximate schemes for solving differential equations is crucial in achieving approximate solutions to the problems at hand.

Recently, the Sub-Ordinary Differential Equation (Sub-ODE) method has been developed as a means for obtaining solutions to nonlinear partial differential equations (NLPDEs). Solutions derived using this method take the form power series based on the solutions of well-established ordinary differential equations (ODEs), such as the Bernoulli and Riccati differential equations. Various types of Sub-ODE methods have been introduced, including the G-expansion, Bernoulli Sub-ODE [13, 23, 24], $(\frac{1}{G'})$-expansion [5, 21], $(\frac{G'}{G})$-expansion [9, 20, 22] and $(\frac{G'}{G}, \frac{1}{G'})$-expansion [10]. A significant advantage of the Sub-ODE method is its ability to produce near-analytical solutions to both ordinary and partial differential equations. This is a primary motivation for this paper, as we propose an approach to derive near-exact solutions, for certain classes of OCPs utilizing the Sub-ODE methods.

The structure of this paper is organized as follows: Section 2 presents the problem of optimal control, while Section 3 offers an overview of Sub-ODE method. In Section 4, we describe how the Sub-ODE method can be applied to solve optimal control problems, including an algorithm that illustrates the application process, along with a discussion on the method's convergence. In Section 5, we demonstrate the effectiveness of the proposed scheme through several numerical examples. The final section concludes the discussion.

2 Preliminaries

Consider the following OCP:

$$\begin{aligned} \text{Minimize} \quad & J = \int_{t_0}^{t_f} \mathcal{F}_\circ(\mathbf{x}, \mathbf{u}, t)dt, \\ \text{s.t.} \quad & \mathcal{F}(\mathbf{x}, \dot{\mathbf{x}}, \mathbf{u}) = 0, \\ & \mathbf{x}(t_0) = \mathbf{x}_0, \end{aligned} \tag{1}$$

where $\mathcal{F}_\circ : \mathbb{R}^{n_x+m_u+1} \to \mathbb{R}$ and $\mathcal{F} : \mathbb{R}^{2n_x+m_u} \to \mathbb{R}^{n_x}$ are polynomial functions. Here, $\mathbf{x}(\cdot) : \mathbb{R}^{n_x} \to \mathbb{R}$ and $\mathbf{u}(\cdot) : \mathbb{R}^{m_u} \to \mathbb{R}$ represent the state and control functions, respectively, both of which are assumed to be continuously differentiable. Also Additionally, $\mathbf{x}_0 \in \mathbb{R}^{n_x}$ is a known vector.

3 An Overview of the Bernoulli Sub-ODE Method

Let w be a function of two variables, x and u. If Q is polynomial in x, u and their derivatives, then the partial differential equation given by

$$Q(w, \frac{\partial w}{\partial x}, \frac{\partial w}{\partial u}, \frac{\partial^2 w}{\partial x^2}, \frac{\partial^2 w}{\partial x \partial u}, \frac{\partial^2 w}{\partial u^2}, \dots) = 0,$$

can be transformed by introducing the new variable $\xi = x + cu$, where c is a nonzero constant .This transformation leads to the ordinary differential equation expressed as

$$P(w, w', w'', w''', \dots) = 0. \tag{2}$$

Consider w to have a power series expansion in terms of ϕ as follows:

$$w(\xi) = \sum_{i=0}^{n} a_i \phi^i(\xi), \tag{3}$$

where ϕ is the solution of the Bernoulli differential equation as defined by

$$\phi' + \lambda\phi = \mu\phi^2. \tag{4}$$

Subsequently, the derivatives of ϕ are computed as follows:

$$\begin{aligned}
&\phi' = -\lambda\phi + \mu\phi^2, \\
&\phi'' = 2\mu^2\phi^3 + \lambda^2\phi, \\
&\phi''' = 2\mu^3\phi^4 - 2\lambda\mu^2\phi^3 + \mu\lambda^2\phi^2 - \lambda^3\phi, \\
&\vdots
\end{aligned}$$

By substituting the expression (3) into (2) and simplifying the resulting expressions, one obtains a polynomial in terms of the various powers of ϕ. By equating the coefficients to zero, a system of algebraic equations can be derived , enabling the determination of the parameters $a_i \ (i = 0, 1, 2, \dots)$.

4 A Parametrization Approach via the Bernoulli Sub-ODE Method

In this section, we consider a control-state parameterization within the governing system of the problem. Substituting the control and state functions in the form of a power series expansion yields:

$$\mathbf{x}(t) = \sum_{i=0}^{n} a_i \Phi^{(i)}(t), \quad u(t) = \sum_{j=0}^{m} b_j \Phi^{(j)}(t), \tag{5}$$

where a_i and b_j are constants for $i = 0, 1, 2, \dots, n$ and $j = 0, 1, 2, \dots, m$. The resulting polynomial is arranged in terms of $\Phi^{(i)} = [\phi_1^i, \dots, \phi_n^i]^T, \quad i = 1, 2, \dots$. To elucidate the parameterization approach through the Bernoulli Sub-ODE method, without losing the generality of the method, we restrict our discussion to the one-dimensional case, i.e., $m_u = n_x = 1$. While noting that this method can be readily generalized to higher dimensions.

For the purposes of this exposition , let us assume that ϕ is the fundamental function used utilized in the parametric representations in Equation (5). Consider the OCP represented as (1), wherein $\mathcal{F}_\circ$ and $\mathcal{F}$ are two polynomials in x, u, and their derivatives.

If x and u can be expressed as finite series in ϕ, according to Equation (5), where a_i and b_j are constants for $i = 0, 1, 2, \dots, n$ and $j = 0, 1, 2, \dots, m$, then m and n are positive integers determined by balancing the highest-order derivative with the highest-order nonlinear term. The function $\phi(t)$ is the solution to the first-order ODE given in Equation (4), expressed in the following form (see [13, 23, 24]):

$$\phi(t) = -\frac{\lambda}{2\mu}\Big(\tanh(\frac{\lambda}{2}t) - 1\Big) \quad (\lambda, \mu \neq 0).$$

To determine n and m, we must identify the two terms with the highest order of ϕ, in the expression

$$\mathcal{F}(\mathbf{x}, \dot{\mathbf{x}}, \mathbf{u}) = 0, \tag{6}$$

subject to the following conditions:

$i.$ At least one of the two terms must be x^i for $i = 1, 2, \dots$ or $\dot{x}$.

$ii.$ One of terms must include u^i for $i = 1, 2, \dots$. If u^i, $i = 1, 2, \dots$ does not appear in in Equation (6), the balance must be established between the expressions $\dot{x}$ and x^j for $i = 1, 2, \dots$ using the highest-order ϕ.

Given these conditions, we can express m in terms of n as $m = g(n)$ where $g : \mathbb{N} \to \mathbb{N}$. The choice of g depends on the specifics of (1). By substituting the power series (5) into (6), simplifying the resulting polynomial in ϕ according to different powers, and setting the coefficients of the various powers of ϕ to zero, we obtain a system of algebraic equations involving the parameters a_i, b_j, λ, and μ. Solving this system leads to the determination of the parameters

a_i, b_j, λ, and μ for $i = 0, 1, 2, \dots, n$ and $j = 0, 1, 2, \dots, m$, allowing us to find the state and control variables outlined in (5) and to ascertain the optimal cost function.

Furthermore, this approach can be extended to control problems expressed in the vector form, where both x and u are vectors. This framework can also accommodate cases involving constraints represented by systems of vector equations. When the exact solution to the OCP is known as an ordered pair of functions $(x^*(t), u^*(t))$ along with the optimal cost function value J^*, the following expressions can be employed to compute the error at each iteration k of the algorithm:

$$\begin{aligned} E_k^{cf} &= \Big| \int_{t_0}^{t_f} \mathcal{F}_\circ(x_k, u_k, t)dt - J^* \Big|, \\ E_k^{c} &= \int_{t_0}^{t_f} \|\mathcal{F}(x_k, u_k, \dot{x}_k)\| dt, \\ E_k^{sc} &= \int_{t_0}^{t_f} |x_k - x^*| dt + \int_{t_0}^{t_f} |u_k - u^*| dt, \\ TE_k &= E_k^{cf} + E_k^{c} + E_k^{sc}, \end{aligned} \tag{7}$$

where the pair $(x_k(t), u_k(t))$ serves as the approximate solution to the OCP. Here, E_k^{cf}, E_k^{c}, E_k^{sc}, and TE_n represent the cost function error, the constraint error, the state-control error, and the total error in the kth iteration, respectively. Based on the preceding discussion , we propose the following algorithm to determine a near-exact solution to the OCP as outlined in Equation (1).

4.1 Convergence analysis

Consider the control and state functions, u, x expanded as detailed in (5). By substituting these functions into the constraints, initial and boundary conditions, and the cost function of the OCP outlined in (1), we obtain the following minimization problem:

$$\begin{aligned} \text{Minimize} \quad & J = \textstyle\int_{t_0}^{t_f} \mathcal{F}_\circ(t, \sum_{i=0}^n a_i \phi^i(t), \sum_{j=0}^m b_j \phi^j(t), \sum_{i=1}^n i a_i \phi^i(t)(\mu\phi(t) - \lambda))dt, \\ \text{s.t.} \quad & \mathcal{F}(\textstyle\sum_{i=0}^n a_i \phi^i(t), \sum_{j=0}^m b_j \phi^i(t), \sum_{i=1}^n i a_i \phi^i(t)(\mu\phi(t) - \lambda)) = 0. \end{aligned}$$

Let $\alpha = (a_0, a_1, \dots, a_n)^T$ and $\beta = (b_0, b_1, \dots, b_m)^T$, then the OCP in (1) can be reformulated as the following optimization problem:

$$\begin{aligned} \text{Minimize} \quad & J = J(\alpha, \beta), \\ \text{s.t.} \quad & N[\alpha, \beta] = 0. \end{aligned}$$

Define S to be the set of all admissible pairs $(x(t), u(t))$, which satisfies the OCP described in (1) along with the specified initial and final conditions.

Algorithm 1 Assume that the maximum number of iterations of the algorithm is k. Consider error thresholds $\epsilon_1 > 0$ and $\epsilon_2 > 0$. Initialize with $n = 0$.

Step 1 Compute the values of n and m using steps (i) and (ii) in Section 4, represented as $m = g(n)$.

Step 2 Increment n by one.

Step 3 Using Equation (5), evaluate the approximate values for x_n and $u_{g(n)}$.

Step 4 Substitute the approximate values x_n and $u_{g(n)}$ into Equation (6) along with the boundary conditions, thereby setting the resulting polynomial coefficients to zero.

Step 5 Solve the system of algebraic equations obtained from the previous step to determine parameters a_i and b_j.

Step 6 To compute x_n and $u_{g(n)}$, substitute the solutions for a_i and b_j into statements derived from Equation (5).

Step 7 Determine the cost function value j_n using the values x_n and $u_{g(n)}$ obtained in the previous step.

Step 8 Calculate the total error TE_n using the expressions provided in Equation (7).

Step 9 If any of the following conditions are met, then the algorithm terminates, otherwise, return to Step 2:

$i)$ If $n \neq 1$, and $|j_n - j_{n-1}| < \epsilon_1$,

$ii)$ $n > k$,

$iii)$ If $TE_n < \epsilon_2$.

Let $S_{n,g(n)}$ represents the set of all functions $(x_n(t), u_{g(n)}(t))$ on the interval $[t_0, t_f]$ such that

$$x_n(t) = \sum_{i=0}^{n} a_i \phi^i(t), \quad u_{g(n)}(t) = \sum_{j=0}^{g(n)} b_j \phi^j(t),$$

which satisfy the dynamic of the problem defined in (1). If there exists a solution to the OCP (1) as expressed in the form of (5), then the convergence of this approach can be characterized by the following theorem.

Theorem 1. Let $r = \inf_S J$, and $r_{n,g(n)} = \inf_{S_{n,g(n)}} J$ for $n = 1, 2, 3, \ldots$. Then, we have

$$\lim_{n \to +\infty} r_{n,g(n)} = r.$$

Proof. Assume

$$r_{n,g(n)} = \min_{(\alpha_n, \beta_{g(n)}) \in \mathbb{R}^{n+1} \times \mathbb{R}^{g(n)+1}} J(\alpha_n, \beta_{g(n)}),$$

then there exists a pair $(\alpha_n^*(t), \beta_{g(n)}^*)$ such that

$$(\alpha_n^*(t), \beta_{g(n)}^*) \in \text{argmin}\{J(\alpha_n, \beta_{g(n)}) : (\alpha_n, \beta_{g(n)}) \in \mathbb{R}^{n+1} \times \mathbb{R}^{g(n)+1}\},$$

and consequently,

$$r_{n,g(n)} = J(\alpha_n^*, \beta_{g(n)}^*).$$

Thus , $(x_n^*(t), u_{g(n)}^*(t)) = (\sum_{i=0}^{n} a_i^* \phi^i(t), \sum_{j=0}^{g(n)} b_j^* \phi^j(t))$ belongs to $S_{n,g(n)}$ and satisfies

$$(x_n^*(t), u_{g(n)}^*(t)) \in argmin\{J(x(t), u(t)) : (x(t), u(t)) \in S_{n,g(n)}\},$$

which indicates that

$$J(x_n^*(t), u_{g(n)}^*(t)) = \min_{(x(t),u(t)) \in S_{n,g(n)}} J((x(t), u(t)),$$

and therefore,

$$r_{n,g(n)} = J(x_n^*(t), u_{g(n)}^*(t)).$$

Noticing the definition of $S_{n,g(n)}$ and increasing function g, it follows that $S_{n,g(n)} \subset S_{n+1,g(n+1)}$. Consequently, we have:

$$\min_{(x(t),u(t)) \in S_{n+1,g(n+1)}} J(x(t), u(t)) \leq \min_{(x(t),u(t)) \in S_{n,g(n)}} J(x(t), u(t)).$$

This implies that the sequence $\{r_{n,g(n)}\}_{n=1}^{+\infty}$ is decreasing. Since this sequence is bounded below, it must converge.

Assuming $\lim_{n \longrightarrow +\infty} r_{n,g(n)} = r^*$ and $r^* > r$, we can set $\varepsilon = \dfrac{r^* - r}{2}$. This leads to the existence of at least one pair $(x(\cdot), u(\cdot)) \in S$ such that $J((x(\cdot), u(\cdot)) < r + \varepsilon < r^*$, which creates a contradiction. Therefore, we conclude that

$$\lim_{n \longrightarrow +\infty} r_{n,g(n)} = \min_{(x,u) \in S} J(x(\cdot), u(\cdot)) = r.$$

□

5 Numerical Results and Discussion

In this section, we evaluate the performance of the algorithm presented in Section 4 through its application to several OCPs.

Example 1. Consider the following OCP (refer to [3, 8, 11, 14]):

$$\begin{aligned} \text{Minimize} \quad & J = \frac{1}{2}\int_0^1 (x^2(t) + u^2(t))dt, \\ \text{s.t.} \quad & x'(t) = -x(t) + u(t), \\ & x(0) = 1. \end{aligned}$$

The analytical solution of this problem, derived using Pontryagin's maximum principle, is as follows [8]:

$$x(t) = Ae^{\sqrt{2}t} + (1-A)e^{-\sqrt{2}t},$$

$$u(t) = A(\sqrt{2}+1)e^{\sqrt{2}t} - (1-A)(\sqrt{2}-1)e^{-\sqrt{2}t},$$

$$J = \frac{e^{-2\sqrt{2}t}}{2}((\sqrt{2}+1)(e^{4\sqrt{2}}-1))A^2 + \frac{e^{-2\sqrt{2}t}}{2}((\sqrt{2}-1)(e^{2\sqrt{2}}-1))(1-A^2),$$

where

$$A = \frac{2\sqrt{2}-3}{-(e^{\sqrt{2}})^2 + 2\sqrt{2} - 3}.$$

In a manner analogous to (5), we express the state and control variables as follows:

$$\begin{aligned} x'(t) &= na_n\phi^{n+1} + \cdots, \\ u(t) &= b_m\phi^m + b_{m-1}\phi^{m-1} + \cdots + b_0. \end{aligned}$$

By equating x' with u, we obtain $m = n+1$ For $n = 13$ and using the initial condition $x(0) = 1$, the coefficients ϕ^i for $i = 0, 1, 2, \ldots, 14$, are derived as follows:

$$\begin{aligned} \phi^0 : &-\frac{1}{8192}\frac{a_{13}\lambda^{13}}{\mu^{13}} - \frac{1}{4096}\frac{a_{12}\lambda^{12}}{\mu^{12}} - \frac{1}{2048}\frac{a_{11}\lambda^{11}}{\mu^{11}} - \frac{1}{1024}\frac{a_{10}\lambda^{10}}{\mu^{10}} - \frac{1}{512}\frac{a_9\lambda^9}{\mu^9} - \frac{1}{256}\frac{a_8\lambda^8}{\mu^8} \\ &- \frac{1}{128}\frac{a_7\lambda^7}{\mu^7} - \frac{1}{64}\frac{a_6\lambda^6}{\mu^6} - \frac{1}{32}\frac{a_5\lambda^5}{\mu^5} - \frac{1}{16}\frac{a_4\lambda^4}{\mu^4} - \frac{1}{8}\frac{a_3\lambda^3}{\mu^3} - \frac{1}{4}\frac{a_2\lambda^2}{\mu^2} - \frac{1}{2}\frac{a_1\lambda}{\mu} - b_0 + 1, \end{aligned}$$

$$\begin{aligned} &\phi^1 : -a_1\lambda + a_1 - b_1 = 0, && \phi^2 : a_1\mu - 2a_2\lambda + a_2 - b_2 = 0, \\ &\phi^3 : 2a_2\mu - 3a_3\lambda + a_3 - b_3 = 0, && \phi^4 : 3a_3\mu - 4a_4\lambda + a_4 - b_4 = 0, \\ &\phi^5 : 4a_4\mu - 5a_5\lambda + a_5 - b_5 = 0, && \phi^6 : 5a_5\mu - 6a_6\lambda + a_6 - b_6 = 0, \end{aligned}$$

$\phi^7 : 6a_6\mu - 7a_7\lambda + a_7 - b_7 = 0,$ $\phi^8 : 7a_7\mu - 8a_8\lambda + a_8 - b_8 = 0,$

$\phi^9 : 8a_8\mu - 9a_9\lambda + a_9 - b_9 = 0,$ $\phi^{10} : 9a_9\mu - 10a_{10}\lambda + a_{10} - b_{10} = 0,$

$\phi^{11} : 10a_{10}\mu - 11a_{11}\lambda + a_{11} - b_{11} = 0,$ $\phi^{12} : 11a_{11}\mu - 12a_{12}\lambda + a_{12} - b_{12} = 0,$

$\phi^{13} : 12a_{12}\mu - 13a_{13}\lambda + a_{13} - b_{13} = 0,$ $\phi^{14} : 13a_{13}\mu - b_{14} = 0.$

By formulating the given OCP transforming it into a non-linear programming problem, we can then solve for the coefficients a_i, b_j, λ, and μ.

When $n = 13$, the optimal cost function and global error are given by

$$J_{13} = 0.192909800170240453,$$

and $e_{13} = 0.00109885557$,respectively.

In Table 1, the cost function and the total error of the approximation for the n-th iteration obtained using Sub-ODE method for $n = 1, 2, \dots, 7$, are presented. Table 2 displays the error values for E_n^{cf}, E_n^c, E_n^{sc} and, E_n^{bc}.

Table 1: Approximate values of the cost function for various value of n

n	Cost function J_n	Total error TE_n
1	0.194979898049381095	0.06566533217
2	0.193399605906644678	0.03222016221
3	0.192938322598352696	0.00763863267
4	0.192926126244441942	0.00594798197
5	0.192918005539758908	0.00426610087
6	0.192916627744976005	0.00400681067
7	0.192913196974128398	0.00290693807

Table 2: Error values in different iterations of the Algorithm 1

n	E_n^{cf}	E_n^c	E_n^{sc}	E_n^{bc}
1	$0.19561333 \times 10^{-2}$	$5.154653423 \times 10^{-10}$	$0.6370919821 \times 10^{-1}$	0
2	0.3758412×10^{-3}	$5.052783997 \times 10^{-10}$	$0.3184432061 \times 10^{-1}$	0
3	0.854421×10^{-4}	$2.232019800 \times 10^{-10}$	$0.755319037 \times 10^{-2}$	0
4	0.976385×10^{-4}	$3.800125559 \times 10^{-10}$	$0.585034307 \times 10^{-2}$	0
5	0.1057592×10^{-3}	$6.004922738 \times 10^{-10}$	$0.416034107 \times 10^{-2}$	0
6	0.1071370×10^{-3}	$1.722175130 \times 10^{-10}$	$0.389967347 \times 10^{-2}$	0
7	0.1105677×10^{-3}	$5.473879389 \times 10^{-10}$	$0.279636967 \times 10^{-2}$	0

Figure 1 illustrates the total error TE_n for various values of n. It is evident that the total error diminishes as n increases.

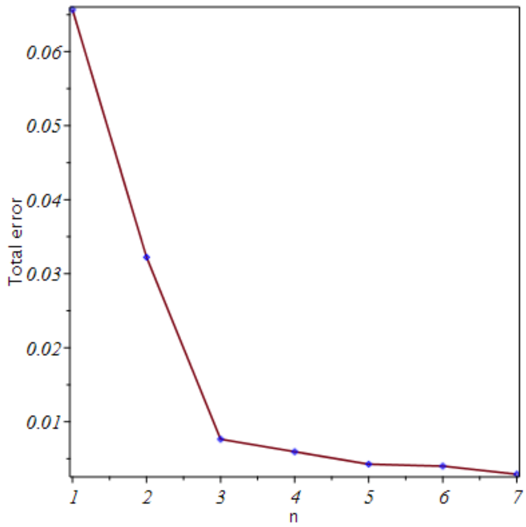

Figure 1: Graph of total error for Example 1.

The comparison of the Sub-ODE method with other methods for calculating the optimal cost function J is presented in Table 3.

Table 3: Comparison of the Sub-ODE method with other methods for computing the optimal cost function J

Method	Cost function J	Absolute error
Sub-ODE method $(n = 13)$	0.192909800170240453	0.1139645×10^{-3}
Salama [14]	0.192909298	0.1144667×10^{-3}
Kafash et al. [8]	0.192909776	0.1139887×10^{-3}
Mehne and Borzabadi [11]	0.193828723	0.8049583×10^{-3}
Elnagar [4]	0.192909281	0.1144837×10^{-3}

Based on the values obtained from the Sub-ODE method in conjunction with other literature , we assert the accuracy and efficiency of the solution derived from the Sub-ODE method given the initial conditions. Figure 2 illustrates the comparison between the exact solutions and those provided by the Sub-ODE solutions for $x(t)$ and $u(t)$.

In Figure 3, we present the error graphs for $x(t)$ and $u(t)$. Finally, the graph depicting the constraint error, $|x'(t) + x(t) - u(t)|$ is shown in Figure 4.

Example 2. In this example, we analyze the following OCP:

$$\begin{aligned} \text{Minimize} \quad & J = \int_0^1 u^2 dt, \\ \text{s.t.} \quad & x'(t) = x^2(t) + u(t), \\ & x(0) = 0, \quad x(1) = 0.5. \end{aligned}$$

To derive the solution through the Sub-ODE method, we express the variables as follows:

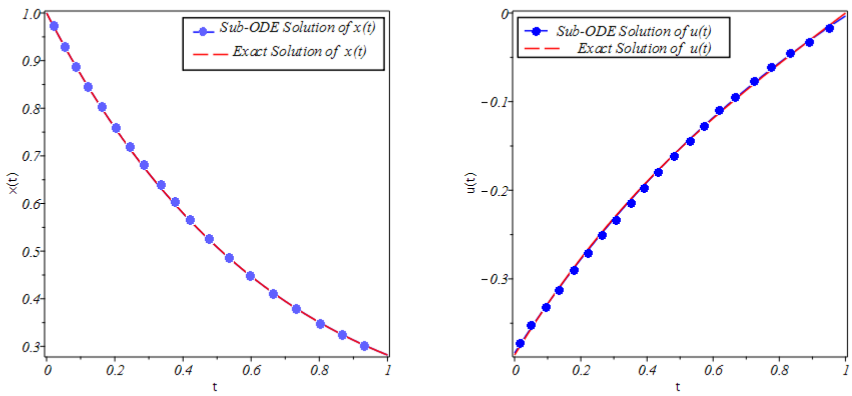

Figure 2: Comparison between the exact solutions and the Sub-ODE solutions of $x(t)$ and $u(t)$ for Example 1.

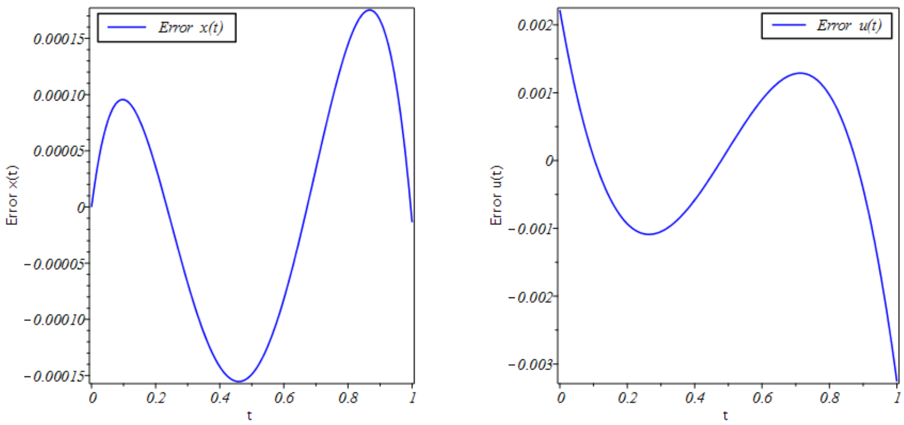

Figure 3: Error graph for $x(t)$ and $u(t)$ for Example 1.

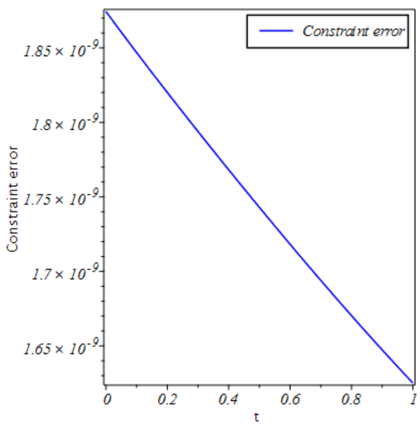

Figure 4: Constraint error graph $|x'(t) + x(t) - u(t)|$ for near-optimal control and state in Example 1.

$$
\begin{aligned}
x^2(t) &= a_n^2 \phi^{2n} + \cdots, \\
u(t) &= b_m \phi^m + b_{m-1} \phi^{m-1} + \cdots + b_0.
\end{aligned}
$$

By balancing the terms x^2 and u, we obtain $m = 2n$. Setting $n = 3$, and applying the initial conditions $x(0) = 1$ and $x(1) = 0.5$, we derive a system of algebraic equations. The coefficients a_i, b_j, λ, and μ are determined as follows:

$$
\begin{array}{lll}
a_0 = -3.132597488, & a_1 = 0.924572439121708, & a_2 = 0.975652406611939, \\
a_3 = -35223.40476, & b_0 = -9.813167025, & \\
b_1 = 5.885492927, & b_2 = 6.489726924, & b_3 = -2.312948918 * 10^5, \\
b_4 = -53263.39757, & b_5 = 68731.59928, & b_6 = -1.240688244 * 10^9, \\
\lambda = -0.100442453339098, & \mu = 1.12042577493801. &
\end{array}
$$

Consequently , the solutions for the state and control variables derived from the Sub-ODE method are presented below:

$$
\begin{aligned}
x(t) = & -3.174039916 - 0.04144242840 \tanh(0.05022122660\, t) \\
& + 0.975652406611939(-0.04482334390 \tanh(0.05022122660\, t) - 0.04482334390)^2 \\
& - 35223.40476(-0.04482334390 \tanh(0.05022122660\, t) - 0.04482334390)^3, \\
u(t) = & -10.07697450 - 0.2638074735 \tanh(0.05022122660\, t) \\
& + 6.489726924(-0.04482334390 \tanh(0.05022122660\, t) - 0.0448233439)^2 \\
& - 2.312948918 * 10^5(-0.04482334390 \tanh(0.05022122660\, t) - 0.04482334390)^3 \\
& - 53263.39757(-0.04482334390 \tanh(0.05022122660\, t) - 0.04482334390)^4 \\
& + 68731.59928(-0.04482334390 \tanh(0.05022122660\, t) - 0.04482334390)^5 \\
& - 1.240688244 * 10^9(-0.04482334390 \tanh(0.05022122660\, t) - 0.04482334390)^6.
\end{aligned}
$$

In Figure 5, the outputs for the state variable $x(t)$ and control variable $u(t)$ obtained via the Sub-ODE method, are depicted. Furthermore, the graph illustrating the constraint error $|x'(t) - x^2(t) - u(t)|$ is presented in Figure 6.

For $n = 3$, the computed cost function is $J = 0.178967094743031296$. Furthermore, both the initial and final conditions have been accurately verified. In Table 4, we present a comparison of the cost function obtained via the Sub-ODE method with those derived from several other methodologies.

The results presented in Table 4 clearly demonstrate the effectiveness of the Sub-ODE method in achieving a lower cost function compared to the other methods.

Example 3. Consider the following OCP with a nonlinear constraint:

$$
\begin{aligned}
\text{Minimize} \quad & J = \int_{0.1}^{\frac{\pi}{2}} (x(t) + u(t))dt, \\
\text{s.t.} \quad & x'(t) = \sqrt{x(t)u(t)},\ t \in [0.1, \tfrac{\pi}{2}], \\
& x(0.1) = 0.009967, \quad u(0.1) = 0.009769.
\end{aligned}
$$

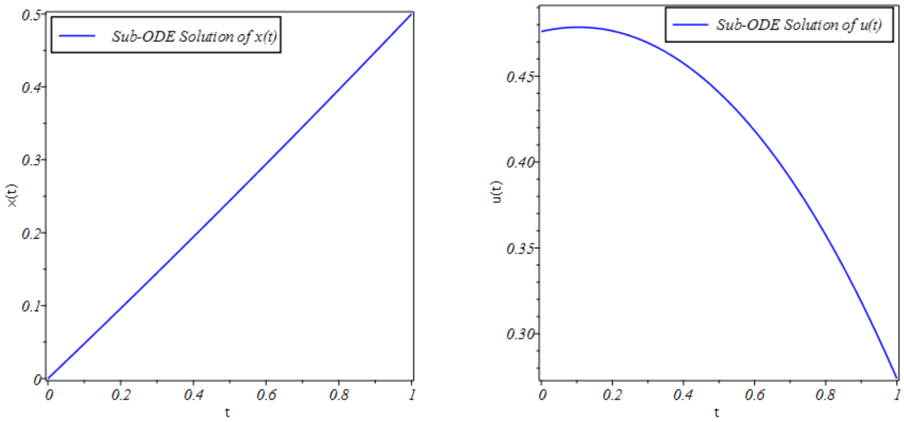

Figure 5: The Sub-ODE solution for the state variable $x(t)$ and the control variable $u(t)$ in Example 2.

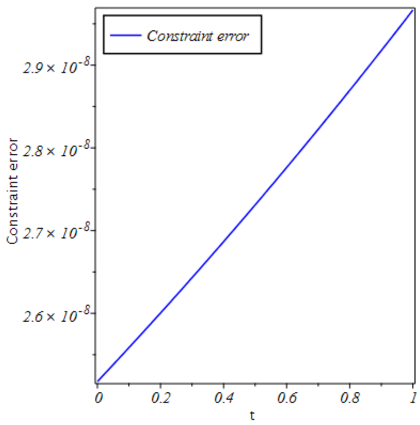

Figure 6: The constraint error graph $|x'(t) - x^2(t) - u(t)|$ for near optimal control and state in Example 2.

Table 4: Comparison of the Sub-ODE method with other methods for computing the optimal cost function J

Method	Cost function J
Sub-ODE $(n = 3)$	0.178967094743031296
Solaymani Fard and Borzabadi [17]	0.4447
Rafiei et al. $(n = 3)$ [12]	0.1791666668

An approximate solution to this problem is presented in [18]. Given the assumption that the control and state functions possess the same sign, the differential equation governing this problem can be transformed as:

$$(x'(t))^2 = x(t)u(t), \quad t \in \left[0.1, \frac{\pi}{2}\right].$$

Using the method outlined in the aforementioned algorithm, we assume that $x(t)$ and $u(t)$ can be expressed in the form detailed in (5), where ϕ is the solution of equation (4). In this context, the expressions $x'(t))^2$ and $x(t)u(t)$ are selected for balancing. We consider the following extensions:

$$(x'(t))^2 = (na_n\phi^{n+1})^2 + \cdots ,$$
$$x(t)u(t) = a_n b_m \phi^{m+n} + \cdots + b_0.$$

After balancing, we establish the relationship $m = n + 2$. Following a single iteration of Algorithm 1, the state function and the control function can be expressed as:

$$x(t) = -0.01197853323 \tanh(1.695339518\, t) + 0.01197853323,$$
$$u(t) = 0.03442841373 - 0.03442841373 \tanh(1.695339518\, t)$$
$$- 0.03442841370(\tanh(1.695339518\, t) - 1)^2 - 0.008607103426(\tanh(1.695339518\, t) - 1)^3.$$

In this instance , the value of the cost function is given by $j_1 = 0.01035959611$. In Table 5, the cost function obtained through the proposed method is compared with those derived from other methods, thus demonstrating the efficiency of this approach. Moreover, Figures 7 and 8 illustrate the state function, control function and the constraint error, respectively.

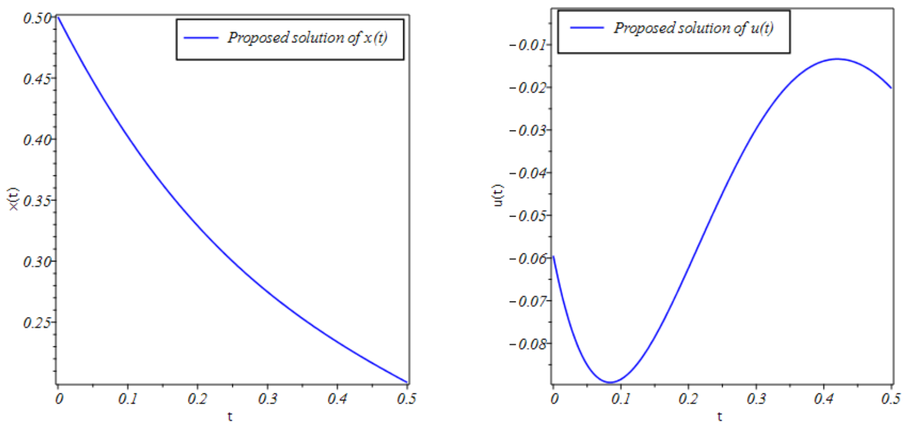

Figure 7: Sub-ODE solutions for the state variable $x(t)$ and the control variable $u(t)$ for Example 3.

Table 5: Comparison of the Sub-ODE method with other methods for computing the optimal cost function J.

Method	Cost function J
Sub-ODE $(n = 1)$	0.01035959611
Tohidi and Saberi Nik [18] $(n = 10)$	0.0290273
Rafiei et al. $(n = 3)$ [12]	0.01116960233

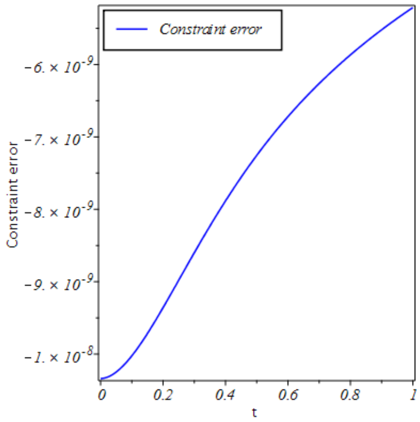

Figure 8: Graph of the constraint error $|(x'(t))^2 - 4x(t)u(t)|$ for near-optimal control and state in Example 3.

Example 4. Consider the following nonlinear OCP:

$$\begin{aligned} \text{Minimize} \quad & J = \int_0^{0.5} (x^2 + u^2) dt, \\ \text{s.t.} \quad & x'(t) = -x(t) - 2x^2(t) - 0.5x^3(t) + u(t), \\ & x(0) = 0.5. \end{aligned}$$

Using the proposed algorithm and expansions as outlined in (5) for both the objective and the control functions, we select the expressions (5) for the objective and control functions, $x^3(t)$ and $u(t)$ for balancing. The expansions will be given as follows:

$$\begin{aligned} x^3(t) &= (a_n)^3 \phi^{3n}(t) + \cdots, \\ u(t) &= b_m \phi^m + \cdots + b_0. \end{aligned}$$

During the balancing step of Algorithm 1, the relationship $3n = m$ will be established. By setting $n = 3$ and applying the algorithm 1, the resulting state and the control functions, along with the constraint error, are depicted in Figures 9, and 10, respectively. In this case, the calculated value of the cost function is 0.0552406197610726332.

Additionally, Table 6 presents the cost function values obtained from the proposed method for $n = 1, 2, 3$ is given. Furthermore, Table 7 compares the objective function derived from the proposed method against other approaches, indicating the efficiency of our method.

Table 6: Approximate values of the cost function for different values of n

n	Cost function J_n	$\lvert j_n - j_{n-1} \rvert$
1	0.0575187620885544079	
2	0.0562647268666681410	0.00125403522
3	0.0552406197610726332	0.00102410711

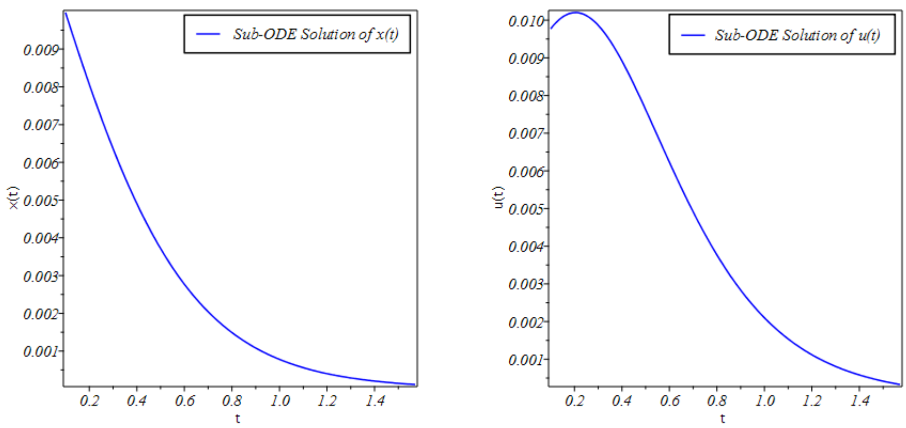

Figure 9: Sub-ODE solutions for the state variable $x(t)$ and the control variable $u(t)$ for Example 4.

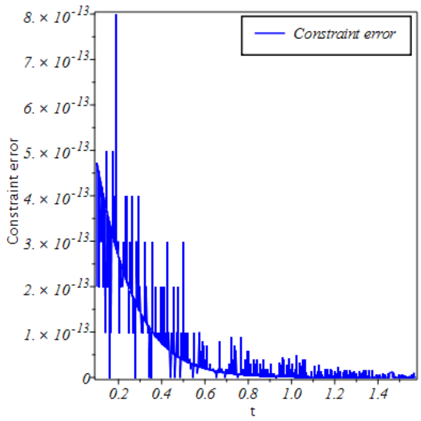

Figure 10: Graph of the constraint error $|x'(t) + x(t) + 2x^2(t) + 0.5x^3(t) - u(t)|$ for near-optimal control and state in Example 4.

Table 7: Comparison of the Sub-ODE method with other methods for computing the optimal cost function J

Method	Cost function J
Sub-ODE $(n = 3)$	0.0552406197610726332
Gholami Baladezaei et al. $(n = 3)$ [6]	0.0552880257005159623
Rafiei et al. $(n = 3)$ [12]	0.0899784856743520989

6 Conclusion

In this paper, we have presented an effective direct scheme for detecting approximate solutions for a specific class of optimal control problems (OCPs). The accuracy of this method was assessed through comparisons with other existing techniques using four numerical examples. Many direct methods based on parameterization require two stages of approximation: one stemming from the solution of the differential equation governing the problem and the other resulting from the optimization problem of the parameterized forms of the state and control variables. In contrast, the approach introduced in this paper, utilizing the sub-ODE method, successfully reduces this process to a single approximation step, arising solely from the resolution of a non-linear optimization problem. While it is crucial to solve the nonlinear optimization problem, associated with the parameterized forms, this study highlights the efficiency of the proposed method by employing the optimization methods available within the Maple software library for all comparative examples. It is important to acknowledge that implementation and algorithmic nature of this method may not be entirely compatible with the methods used for comparison. Therefore, the results presented should not be regarded as definitive evidence of the inadequacies of alternative techniques. Rather, they illustrate the capabilities of the proposed method. The numerical examples demonstrate that the solutions derived using the presented approach are satisfactory. However, it is essential to note that the application of this method may be limited due to the potential need for balancing after the parameterized forms have been incorporated into the governing equations of the system. Addressing this limitation could serve as a compelling area for future research.

Declarations

Availability of Supporting Data
All data generated or analyzed during this study are included in this published paper.

Funding
The authors conducted this research without any funding, grants, or support.

Competing Interests
The authors declare that they have no competing interests relevant to the content of this paper.

Authors' Contributions
The main text of manuscript is collectively written by the authors.

References

[1] Askenazy, P. (2003). "Symmetry and optimal control in economics", Journal of Mathematical Analysis and Applications, 282(2), 603-613.

[2] Edrisi Tabriz, Y., Lakestani, M. (2015). "Direct solution of nonlinear constrained quadratic optimal control problems using B-spline functions", Kybernetika, 51(1), 81-89.

[3] El-Gindy, T.M., El-Hawary, H.M., Salim, M.S., El-Kady, M. (1995). "A Chebyshev approximation for solving optimal control problems", Computers & Mathematics with Applications, 29(6), 35-45.

[4] Elnagar, G.N. (1997). "State-control spectral Chebyshev parameterization for linearly constrained quadratic optimal control problems", Journal of Computational and Applied Mathematics, 79(1), 19-40.

[5] Gachpazan, M., Mortazavi, M. (2015). "Traveling wave solutions for some nonlinear $(N+1)$-dimensional evolution equations by using $(\frac{G'}{G})$ and $(\frac{1}{G'})$-expansion methods", International Journal of Nonlinear Science, 19(3), 137-144.

[6] Gholami Baladezaei, M., Gachpazan, M., Borzabadi, A.H. (2020). "Extraction of approximate solution for a class of nonlinear optimal control problems using $\dfrac{1}{G'}$-expansion technique", Control and Optimization in Applied Mathematics, 5(2), 65-82.

[7] Highfill, J., McAsey, M. (2001). "An application of optimal control to the economics of recycling", SIAM Review, 43(4), 679-693.

[8] Kafash, B., Delavarkhalafi, A., Karbassi, S.M. (2012). "Application of Chebyshev polynomials to derive efficient algorithms for the solution of optimal control problems", Scientia Iranica, 19(3), 795-805.

[9] Kim, H., Sakthivel, R. (2010). "Travelling wave solutions for time-delayed nonlinear evolution equations", Applied Mathematics Letters, 23(5), 527-532.

[10] Li, L.X., Li, E.Q., Wang, M.L. (2010). "The $(\frac{G'}{G}, \frac{1}{G})$-expansion method and its application to travelling wave solutions of the Zakharov equations", Applied Mathematics-A Journal of Chinese Universities, 25(4), 454-462.

[11] Mehne, H.H., Borzabadi, A.H. (2006). "A numerical method for solving optimal control problems using state parametrization", Numerical Algorithms, 42(2), 165-169.

[12] Rafiei, Z., Kafash, B., Karbassi S.M. (2017). "A computational method for solving optimal control problems and their applications", Control and Optimization in Applied Mathematics, 2(1), 1-13.

[13] Salam, A., Sharif Uddin, M.D., Dey, P. (2015). "Generalized Bernoulli sub-ODE method and its applications", Annals of Pure and Applied Mathematics, 10(1), 1-6.

[14] Salama, A.A. (2006). "Numerical methods based on extended one-step methods for solving optimal control problems", Applied Mathematics and Computation, 183(1), 243-250.

[15] Seierstad, A., Sydsaeter, K. (1986). "Optimal control theory with economic applications", Elsevier North-Holland, Inc.

[16] Skandari, M.H.N., Tohidi, E. (2011). “Numerical solution of a class of nonlinear optimal control problems using linearization and discretization”, Applied Mathematics, 2(5), 646-652.

[17] Solaymani Fard, O., Borzabadi, A.H. (2007). “Optimal control problem, quasiassignment problem and genetic algorithm”, 422-424.

[18] Tohidi, E., Saberi, Nik, H. (2015). “A Bessel collocation method for solving fractional optimal control problems”, Applied Mathematical Modelling, 39, 455-465.

[19] Tohidi, E., Navid Samadi, O.R., Farahi, M.H. (2011). “Legendre approximation for solving a class of nonlinear optimal control problems”, Journal of Mathematical Finance, 1(1), 8-13.

[20] Wang, M., Li, X., Zhang, J. (2008). “The $(\frac{G'}{G})$-expansion method and travelling wave solutions of nonlinear evolution equations in mathematical physics”, Physics Letters A, 372(4), 417-423.

[21] Yokus, A. (2011). “Solutions of some nonlinear partial differential equations and comparison of their solutions”, Ph.D. Thesis, Fırat University, Turkey.

[22] Zhang, S., Tong, J.L., Wang, W. (2008). “A generalized $(\frac{G'}{G})$-expansion method for the mKdV equation with variable coefficients”, Physics Letters A, 372(13), 2254-2257.

[23] Zheng, B. (2011). “A new Bernoulli sub-ODE method for constructing traveling wave solutions for two nonlinear equations with any order”, University Politechnica of Bucharest Scientific Bulletin Series A, 73(3), 85-94.

[24] Zheng, B. (2012). “Application of A generalized Bernoulli sub-ODE method for finding traveling solutions of some nonlinear equations”, WSEAS Transactions on Mathematics, 7(11), 618-626.