

Energy-Efficient Adaptive Control for Nonlinear Dynamic Systems

Shiyao Gu

Student in Department of Mathematical Science, University of Liverpool, Liverpool, The United Kingdom

sgsgu4@liverpool.ac.uk

Abstract. This paper focuses on the optimal control of dynamic systems, particularly considering the optimization of reaching target states and minimizing energy consumption. The primary methods employed are Jacobian matrix linearization and state iteration for modeling. Based on this foundation, the paper attempts to solve the optimal control of dynamic systems using optimization problems. Subsequently, the optimization problem is improved by introducing adaptive step sizes and additional constraints to align it more closely with practical applications. The feasibility of these methods is verified from a mathematical perspective. Additionally, the paper examines the impact of constraints on the control problem and the issues that arise when the system approaches boundaries. Through mathematical derivations, the conditions under which constraints prevent the system from converging to the target state are investigated. Overall, this paper presents a novel algorithm for controlling dynamic systems while minimizing energy use. Additionally, it contributes by deriving specific conditions for system convergence under constraints, proposing a trade-off scheme between energy saving and target state achievement, and providing a comprehensive mathematical verification of the method's feasibility.

Keywords: Dynamic system, Optimal control, Jacobian Matrix, State iteration, Convex optimization

1. Introduction

Dynamic systems change their trajectory characteristics over time and with varying control signals, and they have extensive applications in both engineering and natural sciences [1, 2]. In practical applications such as spacecraft control, autonomous driving, and robotics, these systems rely on mathematical principles to achieve specific state targets by formulating reasonable control strategies. These applications often require consideration of various constraints or additional optimization objectives. For instance, in the aviation sector, an aircraft's flight control system must handle complex aerodynamic forces and environmental factors, adjusting control surfaces and engine thrust to maintain stable flight and execute various maneuvers [3]. In autonomous driving technology, vehicles must perceive their environment in real-time and make decisions to adjust speed and direction, ensuring safe and efficient travel [4]. Current research in this area is often characterized by multidimensional and nonlinear dynamics, with the control of nonlinear dynamic systems being a critical theoretical foundation.

In the control of dynamic systems, we typically adjust the system's state through the design of appropriate control signals to ensure it follows a predetermined trajectory [2]. And optimal control theory aims to determine a control strategy that optimizes a performance index, such as energy consumption, time, or error, while satisfying all constraints[5].

Existing approaches to nonlinear optimal control, such as Linear Quadratic Regulation (LQR) and dynamic programming, often face limitations due to the complexities of nonlinear dynamics [6]. These methods generally involve simplifying assumptions or approximations that can lead to suboptimal performance in practical applications. Furthermore, they may not effectively address the challenges posed by real-time constraints and computational complexity.

This paper addresses these gaps by proposing a novel method to optimize control inputs, ensuring the system converges to the desired state trajectory while minimizing energy consumption. The proposed method employs Jacobian matrix linearization and state iteration for modeling, leading to an optimization framework that incorporates adaptive step sizes and additional constraints. The practical significance of this research lies in its potential to enhance control strategies in real-world applications, offering a more efficient and reliable solution for managing nonlinear dynamic systems.

In summary, the main objectives of this paper are to design a control algorithm for a nonlinear system that ensures the system reaches the target state, minimizes control energy, and studies the impact of constraints on the algorithm. We will first provide a mathematical explanation of nonlinear dynamic systems. Then, we will use the Jacobian matrix and state iteration to attempt to model the system. Based on this, we will employ adaptive iterative optimization techniques to balance computational complexity and accuracy, achieving our optimization objectives.

2. Methods

2.1. Mathematical Explanation and Assumptions of the Problem

We will first provide a mathematical explanation of the problem we need to solve and make some necessary assumptions. For a nonlinear system with a theorem in the continuous time domain:

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

where $u(t)$ stands for control signal related to time[7].

Here, we attempt to control the system at discrete time points. We assume that the system's behavior at the current time x_n step depends only on its behavior at the previous time step x_{n-1} and the control signal u_{n-1} (We will explain the rationale behind this approach in the following sections). We also assume that at these specific time points, there exists an explicit function F such that:

$$\text{Here, } \begin{cases} x_{n+1} = F(x_n, U) \\ U = (u_0^T, u_1^T, u_2^T \dots u_N^T) \\ x_n = (x_1^T, x_2^T, x_3^T \dots x_N^T) \end{cases}$$

2.2. Linearization and State Iteration

By the definition of Jacobian Matrix:

$$\delta x_{k+1} = A_k \delta x_k + B_k \delta u_k$$

where $A_k = \frac{\partial f}{\partial x_k}$, $B = \frac{\partial f}{\partial v_k}$

Then with the above formula and assuming the system starts from a stationary position:

$$\begin{aligned} \text{step 1: } \delta x_1 &= B_0 \delta u_0 \\ \text{step 2: } \delta x_2 &= A_1 B_0 \delta u_0 + B_1 \delta u_1 \\ \text{step 3: } \delta x_3 &= A_2 A_1 B_0 \delta u_0 + A_2 B_1 \delta u_1 + B_2 \delta u_2 \\ &\dots \dots \dots \dots \dots \dots \\ \text{step N: } \delta x_N &= A_{N-1} \dots A_1 B_0 \delta u_0 + \dots + B_0 \dots B_{N-1} \delta u_{N-1} \end{aligned}$$

Vectorizing the above equations we get that:

$$H = \begin{pmatrix} B_0 & 0 & \cdots & 0 \\ A_1 B_0 & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ A_{N-1} A_1 \cdots B_0, & A_{N-1} \cdots A_2 B_1 & \cdots & B_{N-1} \end{pmatrix}$$

Thus:

$$\begin{aligned} X_{u+\Delta u} - X_u &= \Delta X = H \Delta u \\ \Rightarrow X_{u+\Delta u} &= X_u + H \Delta u \end{aligned}$$

Here, we are essentially mapping the control signal Δu to the state change ΔX through the methods of linearization and state iteration.

2.3. Solving the Control Problem through the Optimization Problem and the Aforementioned Model

To bring the system state as close to the target state as possible, we decompose the various dimensions of X and use the d_2 distance to measure the difference from the target state. That is:

$$\begin{aligned} \text{minimize}_{\Delta u} \quad & \|x_N + H_N \Delta u - x_{\text{target}}\|^2 + \lambda \|\Delta u\|^2 \\ & H_N \text{ stands for the } n\text{th row of Matrix } H \\ & \lambda \|\Delta u\|^2 \text{ is the regularization term} \end{aligned}$$

Solving this optimization problem completes the first step of our desired solution, which is to bring the system closer to the target state by adjusting the control signal. Next, we will attempt to solve this optimization problem using two methods: one traditional and one improved. In the following section, we will explore the feasibility of this optimization method, as well as approaches for handling some special cases.

2.4. Elementary Strategy to Solve This Optimal Problem

Step1: Apply input u to get A_k and B_k Here we just use our assumption that

$$\begin{aligned} x_{n+1} &= F(x_n, U) \\ \text{and } A_k &= \frac{\partial f}{\partial x_k}, B = \frac{\partial f}{\partial v_k} \end{aligned}$$

Step2: Using the above calculations to get H_N

$$H_N = (A_{N-1} A_1 \cdots B_0, \quad A_{N-1} \cdots A_2 B_1, \quad \cdots, B_{N-1})$$

Step3: Solve the problem to get Δu

$$\text{minimize}_{\Delta u} \quad \|x_N + H_N \Delta u - x_{\text{target}}\|^2 + \lambda \|\Delta u\|^2$$

Step4: Update the control input

$$u \Rightarrow u + \Delta u$$

Step5: Stop iteration if:

$$\|x_N - x_{\text{target}}\| \leq \varepsilon$$

Here is the flowchart for this algorithm[8].

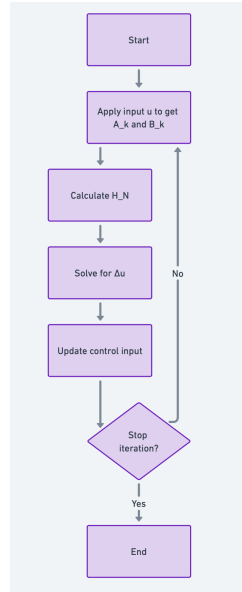


Figure 1. Flowchart of Elementary Strategy

This method provides a prototype for solving this optimization problem. Next, we will make improvements based on this foundation.

2.5. Improved Strategy (Adaptive Step λ) to Solve This Optimal Problem

Now, we improve this method to make it more efficient. Additionally, we introduce some constraints to make it more applicable to real-life problems.

For given target state x_{target} and initial(arbitrary) input u :

We let $\lambda \in (0, 1)$ and our objective function should be: $J(u) = \|X_N(v) - X_{target}\|^2$

Step1: Apply input u to get A_k and B_k

Step2: Using the above calculations to get H_N

Step3: Solve the problem to get Δu

Here, the optimal problem is:

$$\begin{aligned} & \text{minimize}_{\Delta u} \|x_N + H_N \Delta u - x_{target}\|^2 + \lambda \|\Delta u\|^2 \\ & \text{s.t.} \begin{cases} \delta x = H \delta u \\ \delta x_0 = 0 \\ C \delta x \leq d - Cx - \varepsilon \end{cases} \end{aligned}$$

Next, we will explain one by one the significance of each constraint added here.

a) $\delta x = H \delta u$. Indicates that H maps control signal perturbations to state perturbations.

b) $\delta x_0 = 0$. Indicates our assumption of no initial perturbation.

c) $C \delta x \leq d - Cx - \varepsilon$. Here, C represents the mapping from the state space to the constraint space.

We want that:

Before perturbation: $Cx \leq d$, where d represents a constrain in constraint space; And after perturbation: $C(x + \delta x) \leq d$.

Then add a small tolerance ε .

$$\Rightarrow C \delta x \leq d - Cx - \varepsilon$$

This provides an algorithmic example to solve constrained problems, capable of handling some constraint scenarios in practical applications.

Step4: Adjust the iteration step size

Case1: If $J(u + \Delta u^*) < J(u)$, this indicates that the optimization in this step has a positive impact on the objective function. Therefore, we should reduce the iteration step size.

$$\lambda = (1 - a)\lambda$$

Case2: If *otherwise*, this indicates that the optimization in this step has no positive impact on the objective function. Therefore, we should increase the iteration step size.

$$\lambda = (1 + a)\lambda$$

Step5: Update control input

$$u = u + \Delta u^*$$

Step6: Stop iteration if:

$$\|x_N - x_{\text{target}}\| \leq \varepsilon$$

Here is the flowchart for this algorithm.

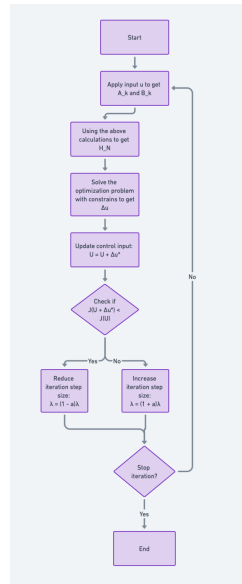


Figure 2. Flowchart of Improved Strategy

Next, we will study some mathematical properties of this optimization problem to verify its feasibility.

3. Relevant Theorem and Mathematical Proofs

For the optimization problem we solved above:

$$\text{minimize}_{\Delta u} \|x_N + H_N \Delta u - x_{\text{target}}\|^2 + \lambda \|\Delta u\|^2$$

which can be generated to:

For a model: $\hat{y} = Ax$, we want to:

$$\min \|y - Ax\|^2 + \lambda \|A\|^2$$

3.1. Lemma: Existence of Forward Optimization

For $\forall u, \exists \delta u^*$, such that:

$$J(u + \delta u^*) \leq J(u) \text{ with } \delta u^* = \operatorname{argmin}_{\delta u} (\|x_N(u) + H_N \delta u - x_{\text{target}}\|^2 + \lambda \|\delta u\|^2)$$

Proof:

We first show that H_N is indeed the derivative of x_N w.r.t. the control signal u .

$$\begin{aligned} \frac{dx_N}{du} \Big|_u &= \left(\frac{\partial x_N}{\partial u_0} \Big|_u, \frac{\partial x_N}{\partial u_1} \Big|_u, \dots, \frac{\partial x_N}{\partial u_{N-1}} \Big|_u \right) \\ &= (A_{N-1}A_1 \cdots B_0, \quad A_{N-1} \cdots A_2B_1, \quad \dots, B_{N-1}) = H_N \end{aligned}$$

Then, as we known, the loss function:

$$J(u + \Delta u^*) = \|x_N(u + \Delta u^*) - x_{\text{target}}\|^2$$

We do first order Taylor Expansion on it:

$$\begin{aligned} \Rightarrow J(u + \Delta u^*) &= \|x_N(u) + H_N \Delta u^* + R(\Delta u^*) - x_{\text{target}}\|^2 \\ &\leq \|x_N(u) + H_N \Delta u^* - x_{\text{target}}\|^2 + \|R(\Delta u^*)\|^2 \\ &\quad + 2 \|x_N(u) + H_N \Delta u^* - x_{\text{target}}\| \cdot \|R(\Delta u^*)\| \end{aligned}$$

$$\text{where } R(\Delta u^*) = [r_1(\Delta u^*), r_2(\Delta u^*) \cdots r_n(\Delta u^*)]^T$$

Then we assume that the coordinate-wise residuals can be upper bounded, that is:

$$\exists M_i, r_i(\Delta u^*) \leq M_i \|\Delta u^*\|^2$$

Take $M = \max M_i$,

$$\|R(\Delta u^*)\|^2 = \sum r_i^2(\Delta u^*) \leq n \left(M \|\Delta u^*\|^2 \right)^2 = nM^2 \|\Delta u^*\|^4$$

Therefore,

$$\begin{aligned} \Rightarrow J(u + \Delta u^*) &\leq \\ \|x_N(u) + H_N \Delta u^* - x_{\text{target}}\|^2 &+ nM^2 \|\Delta u^*\|^4 + 2 \|x_N(u) + H_N \Delta u^* - x_{\text{target}}\| \cdot \sqrt{n}M \|\Delta u^*\|^2 \end{aligned}$$

We know that $\Delta u^* \rightarrow 0$ as $\lambda \rightarrow \infty$.

So that, $\exists \lambda$ s.t.

$$\lambda \geq nM^2 \|\Delta u^*\|^2 + 2\sqrt{n}M \|x_N(u) + H_N \Delta u^* - x_{\text{target}}\|$$

Therefore

$$\begin{aligned} J(u + \Delta u^*) &\leq \|x_N(u) + H_N \Delta u^* - x_{\text{target}}\|^2 + nM^2 \|\Delta u^*\|^4 + \\ 2 \|x_N(u) + H_N \Delta u^* - x_{\text{target}}\| \cdot \sqrt{n}M \|\Delta u^*\|^2 &\leq \|x_N(u) - x_{\text{target}}\|^2 = J(u) \\ \Rightarrow J(u + \Delta u^*) &\leq J(u) \end{aligned}$$

3.2. Theorem: Reachability of the Target State

Assume that $d(x_i, c_i) - \varepsilon > 0$ for $\forall i \in I$. And the model $\delta x_{k+1} = A_k \delta x_k + B_k \delta u_k$ is N-step controllable. Then the iterative control converges. And the system can reach its target:

$$x_N \rightarrow x_{target}$$

First, let's explain a few concepts in this theorem:

$d(x_i, c_i) - \varepsilon > 0$: This is equivalent to a safe distance, meaning the distance from the constraint condition must be greater than ε .

Controllable: $\exists u_0, u_1, \dots, u_{N-1}$, x get to x_N in N steps.

Proof: From *lemma*, we know that:

$$\exists u^* \text{ s.t. } J(u + \Delta u^*) \leq J(u)$$

Now we assume that $u_{\Delta u^*, \lambda}$ is the optimal objective value.

$$\begin{aligned} J(u + \Delta u^*) &\leq J(u) \\ \Rightarrow \{J(u)\} &\text{ is Convergent} \\ \Rightarrow J(u) - u_{\Delta u^*, \lambda} &\text{ converges to } 0 \\ \Rightarrow \Delta u^* &= 0 \\ \Rightarrow H_N^T(x_N - x_{target}) &= 0 \end{aligned}$$

Because that, the linearization is N-step controllable,

$$\Rightarrow H_N \text{ is full rank}$$

$$\Rightarrow x_N \rightarrow x_{target}$$

4. Considerations and Solutions for Subsequent Issues

4.1. Stability Issue of States Approaching Constraints

Here we consider the measures that should be taken when the system state approaches the constraint condition to ensure the system does not exceed the boundaries.

Whenever x_j approaches the constrain[9]:

$$\min_{j=1, \dots, p} d(x_j, c_j) \leq \varepsilon$$

We need this condition:

$$(C_{i, \text{close}} - x_i) \delta x_i \leq 0$$

This condition essentially means that the control direction must be opposite to the direction approaching the constraint. In other words, the system cannot move further towards the constraint direction.

In particular, The vector $(C_{i, \text{close}} - x_i)$ points in the direction approaching the constraint. And the vector δx_i points in the direction of the system's movement. We need to ensure that the angle between them is not less than 90 degrees, i.e. $\theta \geq 90^\circ$.

Thus:

$$\begin{aligned} &(C_{i, \text{close}} - x_i) \delta x_i \\ &= |C_{i, \text{close}} - x_i| \cdot |\delta x_i| \cos \theta \leq 0 \end{aligned}$$

Then, for a nominal input u , we apply the system a slightly perturbed test input:

$$\tilde{u} = u + \Delta \tilde{u}$$

where the small random perturbation is designed by solving the following optimal problem:

$$\begin{aligned} & \text{minimize}_{\delta \tilde{u}} (r_{i,\text{close}} - x_i)^T H_N \delta \tilde{u} \\ & \text{s.t. } \|\delta \tilde{u}\|_\infty \leq \beta \frac{\tilde{d}}{\|H_N\|_\infty} \end{aligned}$$

where $\beta \in (0, 1)$ and $\tilde{d} = \min d(x_i, c_i) - \varepsilon$.

The objective function here $(r_{i,\text{close}} - x_i)^T H_N \delta \tilde{u}$ aims to ensure that the system stays as far from or reduces proximity to the boundary as much as possible. Its principle is similar to what we explained earlier.

The constraint $\|\delta \tilde{u}\|_\infty \leq \beta \frac{\tilde{d}}{\|H_N\|_\infty}$ means that we do not want the system to undergo significant changes, thereby ensuring the stability of the system.

And the solution ensures local dynamic by making $\|\Delta \tilde{x}\|_\infty < \tilde{d}$.

At each step, we store the input and state values obtained from the perturbed trajectory to create a new set.

$$\begin{aligned} D_{\text{new}} &= \{D_{\text{new}}^{\{in\}}, D_{\text{new}}^{\{out\}}\} \\ \text{where, } \begin{cases} D_{\text{new}}^{\{in\}} = \{\tilde{x}_k^T, \tilde{u}_k^T\}^T \\ D_{\text{new}}^{\{out\}} = \tilde{x}_{k+1} \end{cases} \end{aligned}$$

We repeat this process with different values of β .

4.2. Situations where Constraint Conditions Prevent the System from Converging to the Target Position
Here, we study the situations where the constraint conditions prevent the system from converging to the target state, and ultimately derive the conditions under which the constraints prevent convergence.

If the constrain prevent the convergence of the iterative control, then we let:

$$z := x_N - x_{\text{target}}$$

We know that:

$$\begin{aligned} & z \text{ not } \rightarrow 0 \\ & \Rightarrow \text{a trivial solution of the optimal problem} \end{aligned}$$

The optimal problem is:

$$\text{minimize}_{\Delta u} \|x_N + H_N \Delta u - x_{\text{target}}\|^2 + \lambda \|\Delta u\|^2$$

s.t.

$$\begin{aligned} & \delta x_i = H_N \delta u, \forall i = 1, 2, 3 \dots N \\ & I = \{i \mid d(x_i, c_i) \leq \varepsilon\} \end{aligned}$$

That is, the index set of states approaching the constraint
 $(C_{i,\text{close}} - x_i) \delta x_i \leq 0$ for $\forall i \in I$

Then, we transform this problem into an equivalent form of the problem:

$$\text{minimize}_{\Delta u} \frac{1}{2} \|z + H_N \Delta u\|^2 + \frac{1}{2} \|\Delta u\|^2$$

s.t.

$$L \Delta u \leq b$$

Here, $L = N \cdot H$ with N a matrix. And $b = d(x_i, C_i) - \varepsilon$.

We have that:

$$\begin{aligned} & \Delta u^* \text{ is the solution} \\ & \Leftrightarrow 0 \in H_N^T (H_N \Delta u^* + z) + \lambda \Delta u^* + L^T (N_{R^P} (L \Delta u^* - b)) \end{aligned}$$

where $N_\Omega(x)$ is the normal cone of Ω at x . That is:

$$N_\Omega(x) = \{y \mid \forall z \in \Omega, \langle y, z - x \rangle \leq 0\}$$

And let $P = |I|$ (the size of I)

$$\begin{cases} R_-^P = (-\infty, 0]^P \\ R_+^P = [0, +\infty)^P \end{cases}$$

Then we try to find the equivalent condition for $\Delta u^* = 0$.

$$\begin{aligned} & \Delta u^* \text{ is the solution} \\ \Leftrightarrow 0 & \in H_N^T (H_N \Delta u^* + z) + \lambda \Delta u^* + L^T (N_{R_-^P} (L \delta u^* - b)) \\ \Leftrightarrow \exists u \text{ s.t. } 0 & = H_N^T z + L^T u, \quad u \in R_+^P, u^T b = 0, b \in R_P^+ \\ \Leftrightarrow H_N^T z & = L^T (-u), \quad -u \in R_-^P, -u^T b = 0, b \in R_P^+ \\ \Leftrightarrow H_N^T z & \in L^T (R_-^P \cap \{b\}^\perp), \quad b \in R_P^+ \end{aligned}$$

where $\{b\}^\perp$ stands for the orthogonal complement of set $\{b\}$.

Then we let $S := L^T (R_-^P \cap \{b\}^\perp)$, our final conclusion is:

S characterizes the set that could lead to a trivial solution.

Case 1: if $b > 0$, then $S = \{0\}$, which means that all points of the current trajectory reach their critical margin.

Case 2: if $b = 0$, then $S = L^T (R_-^P)$. Then

$$H_N^T (x_N - x_{\text{target}}) \in L^T (R_-^P)$$

this constraint can prevent the convergence[8].

In practical applications, the phenomenon where constraints prevent the system from converging may manifest as overly stringent constraints or conflicts between constraints. For example, in the context of aircraft flight, if the aircraft's range of motion is strictly limited, it may be unable to perform certain maneuvers.

4.3. Optimization Problem for Minimizing Energy

Finally, we present the optimization problem for minimizing energy.

$$\begin{aligned} & \text{minimize}_u \int_0^1 \|u(t)\|^2 dt \\ \text{s.t. } & \begin{cases} \dot{x}(t) = f(x(t), u(t)) \\ x(0) = x_0, \quad x(T) = x_{\text{target}} \\ u \in L^2([0, T], R^m) \end{cases} \end{aligned}$$

where $L^2([0, T], R^m)$ stands for square integrable function from $[0, T]$ to R^m .

Here, we aim to minimize the energy while ensuring that the system follows the specified trajectory. With this, we have addressed all the optimization tasks discussed in this document.

The trade-off between minimizing energy and achieving the target state can be expressed as a multi-objective optimization problem, which can be described as follows:

$$\min_u \int_0^T (\|x(t) - x_{\text{target}}\|^2 + \alpha \|u(t)\|^2) dt$$

where α is a weight parameter used to balance the target state error and energy consumption. By adjusting the value of α , an appropriate balance can be found between achieving the target state and minimizing energy consumption[10].

4.4. Method for Discretizing Continuous Systems

Finally, let's look at how to discretize a continuous system, which also serves as a validation of the feasibility of the methods used earlier in this paper[11].

For a continuous non-linear system:

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

We divide the time into small segments Δt . And in each time segment $t \in [k\Delta T, (k+1)\Delta T]$, we keep the control signal $u(t)$ constant.

$$u(t) \equiv \widetilde{u}_k \text{ for } \widetilde{u}_k \in \mathbb{R}^m$$

Thus, for this non-autonomous system, the state over the time interval $[k\Delta T, (k+1)\Delta T]$ can be represented by the following differential equation:

$$\frac{d}{dt} \begin{pmatrix} \xi_1(t) \\ \xi_2(t) \end{pmatrix} = \begin{pmatrix} f(\xi_1(t), \xi_2(t)) \\ 0 \end{pmatrix}$$

with initial condition:

$$\begin{pmatrix} \xi_1(0) \\ \xi_2(0) \end{pmatrix} = \begin{pmatrix} x(k\Delta T) \\ \widetilde{u}_k \end{pmatrix}$$

This differential equation will define a function $\Phi_{\Delta t}$, s.t.

$$\begin{pmatrix} x((k+1)\Delta T) \\ \widetilde{u}_k \end{pmatrix} = \Phi_{\Delta T} \begin{pmatrix} x(k\Delta T) \\ \widetilde{u}_k \end{pmatrix}$$

Therefore,

$$x_{k+1} = \Phi_{\Delta T}(x_k, u_k) = F(x_k, u_k)$$

We have thus completed the discretization of the continuous system.

5. Discussion

Finally, let's consider some potential limitations of the methods used in this paper. Some of possible shortcomings are as follows:

1. **Constraint Handling:** In practical applications, dynamic systems are often influenced by various complex constraints, such as physical limitations, resource constraints, and operational conditions[2]. In this paper, although we have introduced some basic constraints in the optimization problem to simulate real situations, we have not thoroughly investigated how these constraints affect system performance in more complex or non-standard scenarios. Therefore, future research could consider a wider variety of constraints, especially those that are common in practical applications but not reflected in the model presented in this paper. This will help improve the applicability and robustness of the method in real-world situations.

2. **Discretization Issue:** In control theory, continuous-time systems often need to be discretized for numerical solution and computer implementation[12]. The method proposed in this paper also faces the need for discretization. In our research, we simply discretized the control signals into fixed signals at regular intervals, but this approach may result in some degree of precision loss. Thus, future research

can explore different discretization techniques to better balance computational complexity and system accuracy.

3. Selection of Optimization Parameters: In the method presented in this paper, especially in the adaptive step-size optimization method, some parameters need to be tuned. The specific values of these parameters may require extensive experimentation to determine in different application scenarios. Therefore, researching how to select appropriate parameter values in various applications remains an open question.

4. Subsequent Data Simulation: This paper only provides a theoretical study on the optimal control of nonlinear systems. Although the feasibility of the proposed method has been verified through mathematical approaches, its concrete performance needs to be determined through subsequent data testing.

6. Conclusion

This paper introduces a novel optimization strategy aimed at addressing control problems in nonlinear dynamical systems. Initially, we attempt to model the system using the Jacobian matrix and state iteration methods. For the convergence issue of the system reaching the target state, two optimization methods are proposed. Among them, the more advanced optimization method employs an adaptive step size and takes into account potential constraints in practical applications. Subsequently, through mathematical proofs, we verify the feasibility of this optimization problem. We also consider the conditions under which the model, under certain constraints, may fail to converge and identify these non-convergence conditions. Finally, we introduce an optimization method to achieve optimal energy and explain the implementation of discretizing continuous systems.

In summary, the main contribution of this paper lies in designing an algorithm that balances computational complexity and accuracy to address the control problem of nonlinear systems, while also considering energy optimization. On this basis, the impact of constraints on system convergence is studied, and a detailed mathematical background proof is provided.

References

- [1] Q. Zhu, *Nonlinear systems*. MDPI-Multidisciplinary Digital Publishing Institute, 2023.
- [2] N. U. Ahmed, *Dynamic systems and control with applications*. World Scientific Publishing Company, 2006.
- [3] B. L. Stevens, F. L. Lewis, and E. N. Johnson, *Aircraft control and simulation: dynamics, controls design, and autonomous systems*. John Wiley & Sons, 2015.
- [4] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Transactions on intelligent vehicles*, vol. 1, no. 1, pp. 33–55, 2016.
- [5] D. E. Kirk, *Optimal control theory: an introduction*. Courier Corporation, 2004.
- [6] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, no. 1, pp. 3–20, 2002.
- [7] D. P. Feldman, *Chaos and dynamical systems*. Princeton University Press, 2019, vol. 7.
- [8] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.
- [9] R. Pytlak, *Numerical methods for optimal control problems with state constraints*. Springer Science & Business Media, 1999.
- [10] S. Sharma and V. Kumar, "A comprehensive review on multi-objective optimization techniques: Past, present and future," *Archives of Computational Methods in Engineering*, vol. 29, no. 7, pp. 5605–5633, 2022.
- [11] A. A. Marouf and S. A. Al-Assadi, "Computer-aided discretization of continuous data control systems," *Computer-aided design*, vol. 17, no. 4, pp. 169–178, 1985.
- [12] A. C. Luo and A. C. Luo, "Discretization of continuous systems," *Discretization and Implicit Mapping Dynamics*, pp. 51–158, 2015.