Finite-Time Stabilization-Based Adaptive Fuzzy Control Design

Prepered for Adaptive Control Project
Professor Bagheri

Presentation
Murtaza Asaadi

Summer 2025

Outline

What is the Problem?

- Controlling nonlinear systems is challenging due to uncertainties and complex dynamics.
- Traditional controllers guarantee stability over an infinite time horizon (asymptotic stability).

Why Finite-Time Control?

Many real-world applications (robotics, aerospace) require systems to reach their desired state within a specific, finite time.

Benefits of Finite-Time Stability:

- Faster convergence rates.
- Higher precision in tracking.
- Improved robustness against disturbances.

Contribution

- Develops an adaptive fuzzy control strategy for a class of nonlinear systems.
- Guarantees that the system's tracking error converges to a small region around zero in finite time.
- Ensures all signals in the closed-loop system remain bounded and stable.

System Representation

$$\begin{cases} \dot{z}_i = \phi_i(\bar{z}_i) + \varphi_i(\bar{z}_i) z_{i+1}, \quad i = 1, \ldots, n-1 \\ \dot{z}_n = \phi_n(z) + \varphi_n(z) q(v) \\ y = z_1 \end{cases}$$

- z_i: System states.
- ϕ_i, φ_i: Unknown nonlinear functions.
- v: The control input we design.
- $q(v)$: A quantizer, which models the digital nature of controllers (discrete signal levels).
- y: The system output.

Control Objective

Design a **control law** v such that the **system output** y follows a desired **reference signal** y_r in finite time, despite the unknown functions and the quantizer.

Key Components of Control Strategy

The controller is designed using a combination of techniques:

- Backstepping
- Fuzzy Logic Systems (FLS)
- Adaptive Control
- Hysteretic Quantizer

Backstepping

A recursive design methodology. It breaks down the complex n-dimensional system into a series of 1-D problems, designing a "virtual controller" at each step.

Used as universal approximators. Since the functions ϕ and ϕ_i are unknown, an FLS is used to estimate their behavior online.

The parameters of the Fuzzy Logic System are not fixed; they are "adapted" or tuned in real-time by adaptive laws to improve approximation accuracy.

Hysteretic Quantizer

The control input v is passed through a quantizer $q(v)$. This models the constraints of digital hardware and communication channels.

Controller and Adaptive Law Design

The design follows the backstepping procedure, step-by-step.

- Step 1: Virtual Controller α_1
- Step 2: Actual Controller v
- Adaptive Laws

Controller and Adaptive Law Design

The design follows the backstepping procedure, step-by-step.

- Step 1: Virtual Controller α_1
  - Define the first error surface: $\eta_1 = z_1 - y_r$.
  - Design a virtual controller α_1 to stabilize this error.
  - α_1 includes terms to drive the error to zero and a fuzzy logic term to cancel nonlinearities.
- Step 2: Actual Controller v
- Adaptive Laws

Controller and Adaptive Law Design

The design follows the backstepping procedure, step-by-step.

- Step 1: Virtual Controller α_1
- Step 2: Actual Controller v
  - Define the final error surface: $\eta_2 = z_2 - \alpha_1$.
  - Design the actual control input v to stabilize η_2.
- Adaptive Laws

Controller and Adaptive Law Design

The design follows the backstepping procedure, step-by-step.

- Step 1: Virtual Controller α_1
- Step 2: Actual Controller v
- Adaptive Laws
  - For each fuzzy system, an adaptive law updates its weight parameter θ_i.
  - The goal is to minimize the function approximation error.

A total Lyapunov function V is constructed for the entire closed-loop system. The paper proves that the derivative of the Lyapunov function satisfies:

$$\dot{V} \leq -\lambda_0 V - \lambda_1 V^h + b_0$$

- $\lambda_0 V$: Guarantees exponential stability.
- $\lambda_1 V^h$: Guarantees finite-time stability.
- b_0: A small positive constant due to approximation errors.

Theorem 1 (Main Result)

The designed controller ensures that:

- The system is practically stable in finite time. The tracking error converges to a small, bounded region around zero.
- The size of this region and the convergence time can be calculated.
- All signals within the system (states, adaptive parameters) remain bounded.

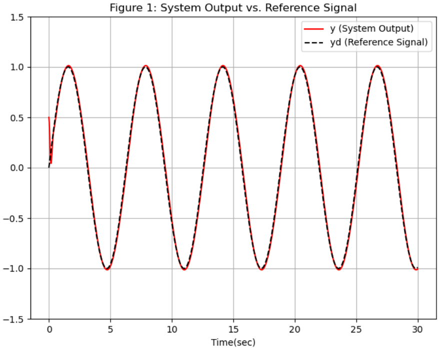
Python Implementation Overview

The simulation was reproduced using Python with SciPy and Matplotlib.

Key Implementation Steps:

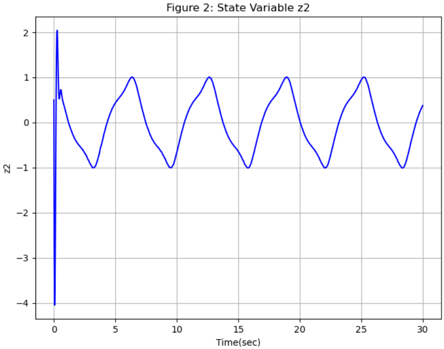
- Define Parameters
- Hysteretic Quantizer
- Fuzzy Basis Functions
- System Model Function
- ODE Solver
- Plotting

System Output vs. Reference Signal



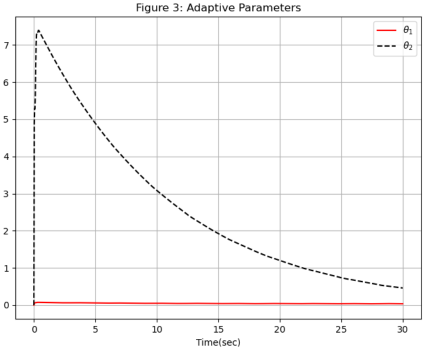
- The plot shows the system output y quickly converging to and tracking the sinusoidal reference signal y_d.
- This demonstrates the effectiveness of the tracking control.

State Variable z_2



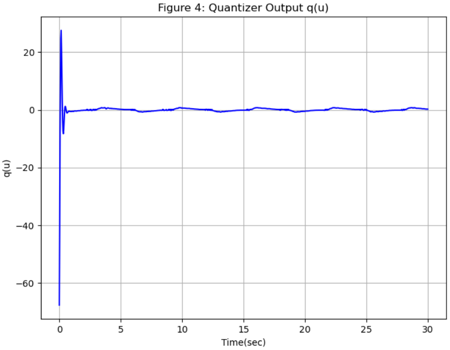
- The state z_2 remains bounded throughout the simulation, confirming the stability of all system signals.

Adaptive Parameters



- The adaptive parameters θ_1 and θ_2 converge to stable values.
- This indicates that the fuzzy logic systems have successfully learned to approximate the unknown nonlinearities.

Quantizer Output



- This plot shows the discrete output of the hysteretic quantizer.
- It highlights that the controller operates effectively even with a non-continuous, quantized input signal.

Conclusion

- Successfully simulated an adaptive fuzzy controller that achieves finite-time stability for a class of nonlinear systems.
- The use of backstepping and fuzzy logic systems effectively handles system uncertainties.
- The design explicitly accounts for input quantization, making it more practical for digital implementation.
- Stability analysis proves that all signals are bounded and the tracking error converges to a small region in finite time.
- The Python simulation verifies the theoretical results, showing excellent tracking performance and stability.

Conclusion

- Successfully simulated an adaptive fuzzy controller that achieves finite-time stability for a class of nonlinear systems.
- The use of backstepping and fuzzy logic systems effectively handles system uncertainties.
- The design explicitly accounts for input quantization, making it more practical for digital implementation.
- Stability analysis proves that all signals are bounded and the tracking error converges to a small region in finite time.
- The Python simulation verifies the theoretical results, showing excellent tracking performance and stability.

Conclusion

- Successfully simulated an adaptive fuzzy controller that achieves finite-time stability for a class of nonlinear systems.
- The use of backstepping and fuzzy logic systems effectively handles system uncertainties.
- The design explicitly accounts for input quantization, making it more practical for digital implementation.
- Stability analysis proves that all signals are bounded and the tracking error converges to a small region in finite time.
- The Python simulation verifies the theoretical results, showing excellent tracking performance and stability.

Conclusion

- Successfully simulated an adaptive fuzzy controller that achieves finite-time stability for a class of nonlinear systems.
- The use of backstepping and fuzzy logic systems effectively handles system uncertainties.
- The design explicitly accounts for input quantization, making it more practical for digital implementation.
- Stability analysis proves that all signals are bounded and the tracking error converges to a small region in finite time.
- The Python simulation verifies the theoretical results, showing excellent tracking performance and stability.

Conclusion

- Successfully simulated an adaptive fuzzy controller that achieves finite-time stability for a class of nonlinear systems.
- The use of backstepping and fuzzy logic systems effectively handles system uncertainties.
- The design explicitly accounts for input quantization, making it more practical for digital implementation.
- Stability analysis proves that all signals are bounded and the tracking error converges to a small region in finite time.
- The Python simulation verifies the theoretical results, showing excellent tracking performance and stability.

Conclusion

- Successfully simulated an adaptive fuzzy controller that achieves finite-time stability for a class of nonlinear systems.
- The use of backstepping and fuzzy logic systems effectively handles system uncertainties.
- The design explicitly accounts for input quantization, making it more practical for digital implementation.
- Stability analysis proves that all signals are bounded and the tracking error converges to a small region in finite time.
- The Python simulation verifies the theoretical results, showing excellent tracking performance and stability.

Thank you.
Any questions?