LSTM 기반 비선형 모델 예측 제어기의 성능 해석

알렉산더 마이어, 김정훈
포항공과대학교 전자전기공학과 (alexmeyer@postech.ac.kr, junghoonkim@postech.ac.kr)

Performance Analysis Of LSTM-Based Nonlinear Model Predictive Controllers

Alexander Meyer, Jung Hoon Kim
Department of Electrical Engineering, POSTECH (alexmeyer@postech.ac.kr, junghoonkim@postech.ac.kr)

Abstract - This paper examines the performance of Long Short-Term Memory (LSTM) networks as a replacement for Nonlinear Model Predictive Controllers (NLMPCs) in plants with fast dynamics that usually cause controller instability. Equations of motion are used to model plant dynamics for a inverted rotary pendulum, also known as the Furuta pendulum. In this experiment, we produce training data for the LSTM by observing the NLMPC actively computing control moves to stabilize the pendulum. After training, the LSTM is assigned to control the system independently. Simulated results show the mean computation time required for a control move is two orders of magnitude lower in the LSTM than that of the NLMPC while maintaining similar set-point tracking performance characteristics.

1. Introduction

Nonlinear systems are notoriously difficult to control, and most successful solutions are system dependent. Nonlinear Model Predictive Control (NLMPC) has seen success in traversing the state space of many nonlinear plants but is limited to systems with a response time on the order of seconds. Controller response times in the process industry tend to be greater than one second, and NLMPC is often successfully deployed to plants, such as continuously stirred tank reactors. However, when plant dynamics require response times lower than 100 ms, NLMPC is highly unlikely to be capable of stably controlling the system [5].

Several techniques exist to address the problem of controlling plants with fast dynamics, such as linearizing around operating points, Reinforcement Learning, or Explicit MPC. The upside of using NLMPC over these methods is the generalizable structure, which can handle constraints and manage performance characteristics by tuning weights. NLMPC is also robust in the face of time delays, noise, and unmeasured disturbances.

Long Short-Term Memory networks are machine learning models that examine sequences and produce a prediction for the following item in the sequence. They are naturally suited to using time-series data and rejecting time delays and do not suffer from the so-called *vanishing gradient* problem that plagues many recurrent models in the field.

2. Main

2.1 Overview

The goal of the experiment is to stabilize the inverted rotary pendulum in the upright position with multiple controllers. To achieve this, we first define the model and establish a control scheme, then generate training data for the LSTM, and finally re-deploy both controllers for quantitative comparison.

2.2 Pendulum Model

The lowest-energy configuration of the pendulum is defined as π and rotation of the bodies about their pivot is positive when counter-clockwise (Fig. 1). Equations of motion (1) are derived by applying the Lagrangian to energy expressions of each system component [1], and defining the inertial terms separately, for clarity (2). Values in Table 1 are extracted from the Quanser QUBE Servo 2, a commercially available inverted rotary pendulum, making this simulated system realistically model a known physical system. Although included, Coulomb friction was ignored, and viscous friction was only included for the motor interface (3).

$$\begin{aligned}(\alpha+\beta\sin^2\theta)\,\ddot{\phi}+\gamma\cos\theta\ddot{\theta}+2\beta\cos\theta\sin\theta\dot{\phi}\dot{\theta}-\gamma\sin\theta\dot{\theta}^2&=\tau_\phi\\ \gamma\cos\theta\ddot{\phi}+\beta\ddot{\theta}-\beta\cos\theta\sin\theta\dot{\phi}^2-\delta\sin\theta&=\tau_\theta\end{aligned} \tag{1}$$

where,

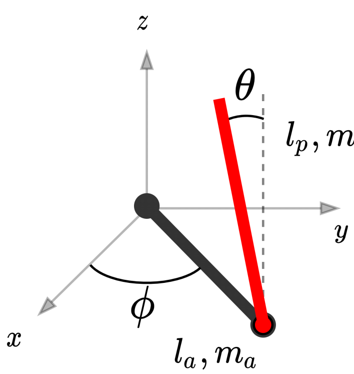
$$\begin{aligned}\alpha &\triangleq J+(M+\tfrac{1}{3}m_a+m_p)l_a^2 & \beta &\triangleq (M+\tfrac{1}{3}m_p)l_p^2\\ \gamma &\triangleq (M+\tfrac{1}{2}m_p)l_a l_p & \delta &\triangleq (M+\tfrac{1}{2}m_p)gl_p\end{aligned} \tag{2}$$

External torque on the pendulum is considered a disturbance, and the torque on the rod is the control variable produced by the motor. Friction is modeled as a force that directly resists the input torque:

$$\tau=\tau_u-\tau_F\,, \qquad \tau_F=\tau_C\operatorname{sgn}\dot{\phi}+\tau_v\dot{\phi} \tag{3}$$

<Table 1> Model Parameters

Parameter	Value	Unit
J	4.6×10^{-6}	$kg \cdot m^2$
m_a	0.095	kg
m_p	0.024	kg
l_a	0.085	m
l_p	0.129	m
g	9.81	m/s^2
τ_v	0.001	N

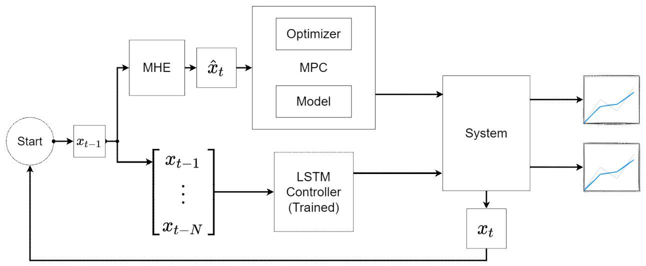


<Figure 1> Inverted rotary pendulum coordinate system diagram

2.3 Generating Training Data

Data used to train the behavior of the LSTM is generated by allowing the NLMPC to stabilize the system in 360 simulations, each with different initial conditions randomly distributed within $[-2\pi<\theta<2\pi]$. The intent here was to provide the LSTM an adequate description of the model state space. Gaussian white noise was included to harden the LSTM against fluctuating sensor readings.

During experimental trials, it was observed that without a diverse set of initial conditions, the LSTM had poor robustness to noise; unless the starting position of the pendulum was identical to that of the training data, the LSTM was operating outside of the range of what it had seen in training.



<Figure 2> Block diagram showing the comparison method at runtime used for the experiment

2.4 Analysis

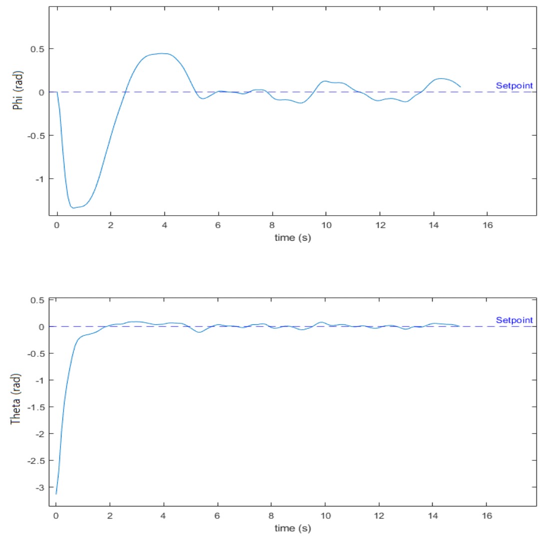
Both controllers were tasked with stabilizing the pendulum upright $(\theta=0)$ from a distributed range of initial states $[\pi/2<\theta<3\pi/2]$ over ten trials, often referred to as "*swing-up control*" when discussing inverted pendulum systems. The LSTM has five layers with 100 neurons each and uses a variable input sequence length of up to 30. The NLMPC has a sampling time of 20 ms, which is highly uncharacteristic and traditionally unstable. The prediction horizon p is 30 steps, which allows the controller to optimize over a 600 ms period. The control horizon m is 5. Quantitative experimental results can be summarized in Table 2.

It is interesting to note that the LSTM has learned to prioritize stabilization of the pendulum over the arm, reaching a steady-state faster than the NLMPC but delaying stabilization of the arm (Fig. 3).

The quintessential reason the NLMPC successfully computed control moves during the simulation was that the system dynamics *paused* until the controller could finish the quadratic programming (QP) optimization problem. NLMPC, during this simulation, was solving the QP problem online using a method similar to *active set,* called *sequential quadratic programming* [2],[3]. It should be noted that there are many methods to customize and tune the performance of the NLMPC, such as changing the QP solver to an *interior point* algorithm [4]. This study has a restricted scope that utilizes standard optimization techniques that a typical engineer would deploy in industry, using common control software.

<Table 2> Controller Performance Analysis for Swing-up Control

Characteristic	LSTM	NLMPC
Max Move Time	39 ms	3440 ms
Mean Move Time	32 ms	380 ms
Constraint Violation	No	No
Steady State (Arm)	4.8 s	2.3 s
Steady State (Pend)	1.2 s	2.4 s
Cost (Arm)	44.0	4.8
Cost (Pend)	12.9	64.4



<Figure 3> Trained LSTM tracking controller performance over 15 seconds for a *swing-up control* simulation with added Gaussian white noise

3. Conclusion

LSTM-based networks have shown to be robust in noise rejection, set-point tracking, and stability, comparable to that of an NLMPC. However, computational control effort has been substantially decreased in the LSTM to the point where the controller is realizable for the inverted rotary pendulum and other plants with fast dynamics, a feat known to be too challenging for an NLMPC [5]. There are several methods available to further this research, such as incorporating disturbance rejection into the training data and deploying the LSTM to the physical system.

[Acknowledgments]

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT). (No. 2022R1F1A1 074251).

[Reference]

[1] Gäfvert M. Modelling the Furuta Pendulum. Department of Automatic Control, Lund Institute of Technology (LTH), 1998.

[2] Gill, P. E., W. Murray, and M. H. Wright. "Numerical Algebra and Optimization, Vol. 1." (1991).

[3] Powell, M.J.D., "A Fast Algorithm for Nonlinearly Constrained Optimization Calculations," Numerical Analysis , G. A. Watson, Lecture Notes in Mathematics, Springer Verlag, Vol. 630, 1978

[4] Wright, Stephen J. Primal-dual interior-point methods. Society for Industrial and Applied Mathematics, 1997.

[5] MathWorks Inc, Generate Code And Deploy To Real-Time Targets, Sampling Rate in Real-Time Environment. 2022.