Discussion on Lyapunov-based Nonlinear Programming (LBNLP)

Kyunghwan Choi, Ph.D.

Assistant Professor
School of Mechanical and Robotics Engineering
Gwangju Institute of Science and Technology (GIST)

15 October 2024

GIST | SMRE 광주과학기술원 기계로봇공학부

Review Comments

- **Reviewer 4 (Review371)**
  - 1. In (27), the selection of β; when maximum eigenvalue of the hessian (HL) is not correct because it cannot satisfy the inequality (23) which is the necessary condition to guarantee dV/dt < 0. Please explain this.
  - 2. The authors claim that there are no tuning parameters in the presented work (Table 1). However, the parameter is the tuning parameter which is still needed to be defined by the user. Please clarify this.
  - 3. Please clearly mention which MPC settings have been used during validations.
  - 4. Following previous comment, the authors claim that usual a prediction horizon of N = 1 is sufficient . Please elaborate more on this and justify it in the text (not just through references). More importantly, setting N=1 would result in a poor prediction. Also, considering that the length of the optimization variables is short with N=1, we already expect to not to have a high computational time. Please clarify this.
  - 5. Please comment on the computational time of the proposed work in comparison to the authors' previous work in [9].
  - 6. Which solver has been used to solve the QPs in the SQP formulation?

Theory part에서는 PD라고 assumption하고 application에서 target system의 파라미터로 표현 시 PD하다는 것을 보여주는 것도 방법이 될 수 있음

Review Comments

- **Reviewer 8 (Review379)**
  - 1. The review of Lyapunov theory at the end of page 3 is not correct for several reasons. First it is not specified that the function defined in (17) is positive definite with respect to any point x_e, namely it is not said where the function is zero and that for all other x's it is strictly positive, as it is required by Lyapunov's method. Moreover, if the condition above holds then ₩dot{V} strictly less than zero then it can be claimed that x_e (not "the system", which is not even defined therein) is asymptotically stable, rather than stable.
  - 2. Similarly to the previous concern, also the fact that the last line of (26) is indeed negative definite for any x is not discussed at all. As a matter of fact, it is stated that the derivative is in fact only negative semi-definite, hence somewhat contradicting the previous review of Lyapunov theory. Before (28a) the Authors say "P is positive definite": this claim should be motivated.
  - 3. In equation (12), the matrix-valued function is defined to belong to R^{n ₩times n_c}, while before (15) it becomes a square matrix in R^{n ₩times n}. This point must be clarified, even if it is just a clash of notation.
  - 4. The Authors should verify whether the inequality sign in (6c) is indeed correct or if it is a typo.

Feasible point에서 시작하면 해결됨. IPM의 feasible initial point 선택하는 방법 참고

Review Comments

- **Reviewer 12 (Review33909)**
  - 1. In the reformulation of NMPC problem (30) to (31), why is the final tracking error adopted as an equality constraint while the tracing errors in other steps are ignored.
  - 2. From the manuscript, it appears that the solution of the proposed approach may violate the equality constraint. If applied in NMPC, will it adversely affect the safe operation of the controlled object?
  - 3. In the case study, although the proposed approach is less computationally expensive than the ALM method, both algorithms take microseconds to compute, while the ALM method is more accurate. Does this mean that the proposed method can only show advantages in control problems that are more computationally expensive?
  - 4. In Figs. 1-4, please indicate the meaning of various symbols to make them easier for readers to understand.

Review Comments

- **Reviewer 17 (Review33919)**
  - 1. Why function "f" and equality and inequality constraints need to be "at least" continuously differentiable, since authors claim there method manages more than only convex problems?
  - 2. The use of lyapunov function to ensure the algorithm convergence is very interesting.
  - 3. In the update law 2 (proposed) the authors claim that"inverse of H (x, λ L A) exists but does not need to be known" and the end of the proof, authors state "Please note that, for a proper choice of β, rough knowledge of HL is required." This is a clear contradiction that needs to be discussed.
  - 4. Comparison of the proposed LBNLP with other methods(SQP, and ALM) is appreciated and clarify the main contribution.
  - 5. Simulations proposed by authors on a classical problem provide an interesting benchmark comparison for the LBNLP method new proposed.
  - 6. Overall the paper is well written and introduce a new approach (even if it is (Lyapunov) well known in the control community) to solve the NLP problem.

Review Comments

- **Reviewer 19 (Review33923)**
  - 1. Please specify the minimizer of the Lyapunov function candidate (17). Is the minimizer of this Lyapunov function (0,0) or (x*, lambda*)? If it is the latter, the statement about Lyapunov second method should have been or all (x, lambda) non equal to (x*, lambda*). Please can the authors explain more on why a stronger convergence can also be achieved?
  - 2. Finally, some typos should be revised, e.g. In Sec II-A, the explanations for mathbb{I} and mathbb{E} should have been in the opposite way.

Improvements needed based on review comments

- Demonstrate LBNLP with an NMPC problem with a longer prediction horizon.
- Specify the minimizer for the Lyapunov function and verify it is positive definite.
- Validate (instant) constraint violations.

- Try another conference paper? Or Submit to a journal with experimental data?

1) 모터제어에 특화해서 ECCE나 IECON에 제출
2) 실험추가해서 저널 제출
3) 일반화해서 또 다른 저널 제출

Extension of Lyapunov-based approach

- **1. Constrained Optimization-Based Neuro-Adaptive Control (CoNAC)**

Under review for *IEEE Transactions on Neural Networks and Learning Systems*

Constrained Optimization-Based Neuro-Adaptive Control (CoNAC) for Uncertain Euler-Lagrange Systems Under Weight and Input Constraints

Myeongseok Ryu, Donghwa Hong, and Kyunghwan Choi, *Member, IEEE*

***Abstract*—This study presents a constrained optimization-based neuro-adaptive controller (CoNAC) for uncertain Euler-Lagrange systems subject to weight norm and input constraints. A deep neural network (DNN) is employed to approximate the ideal stabilizing control law, compensating for lumped system uncertainties while addressing both types of constraints. The weight adaptation laws are formulated through a constrained optimization problem, ensuring first-order optimality conditions at steady state. The controller's stability is rigorously analyzed using Lyapunov theory, guaranteeing bounded tracking errors and DNN weights. Numerical simulations comparing CoNAC with three benchmark controllers demonstrate its effectiveness in tracking error regulation and satisfaction of constraints.**

***Index Terms*—Neuro-adaptive control, constrained optimization, deep neural network, Euler-Lagrange system, input constraint.**

I. INTRODUCTION

A. Background

MANY engineering systems, including those in aerospace, robotics, and automotive applications, can be modeled using Euler-Lagrange systems. These systems are governed by dynamic equations derived from energy principles and describe the motion of mechanical systems with constraints. In practice, however, such systems often exhibit uncertainties due to unmodeled dynamics, parameter variations, or external disturbances. These uncertainties can significantly degrade control performance and, in some cases, lead to instability. To address these challenges, adaptive control methods have been widely employed to ensure robust performance in the presence of system uncertainties [1], [2].

More recently, neuro-adaptive control approaches have been introduced to approximate unknown system dynamics or entire control laws using neural networks (NNs) [3]. NNs are well-known for their universal approximation property, which allows them to approximate any smooth function over a compact set with minimal error. Various types of NNs have been utilized in neuro-adaptive control, including simpler architectures like single-hidden layer (SHL) neural networks [4], [5] and radial basis function (RBF) neural networks [6], [7], as well as more complex models like deep neural networks (DNNs) [8] and their variations. SHL and RBF NNs are often employed to approximate uncertain system dynamics or controllers due to their simplicity [5], [9]–[11], while DNNs offer greater expressive power, making them more effective for complex system approximations [12]. Additionally, variations of DNNs, such as long short-term memory (LSTM) networks for time-varying dynamics [13] and physics-informed neural networks (PINNs) for leveraging physical system knowledge [14], have further extended the capabilities of neuro-adaptive control systems.

A critical aspect of neuro-adaptive control is the weight adaptation law, which governs how NN parameters are updated. Most studies derived these laws using Lyapunov-based methods, ensuring the boundedness of the tracking error and weight estimation error, thus maintaining system stability under uncertainty.

However, two significant challenges persist in using NNs for adaptive control. First, the boundedness of NN weights is not inherently guaranteed, which can result in unbounded outputs. When NN outputs are used directly in the control law, this may lead to excessive control inputs, violating input constraints. Such constraints are commonly encountered in industrial systems, where actuators are limited by physical and safety requirements in terms of amplitude, rate, or energy [15]. Failing to address these constraints can degrade control performance or even destabilize the system.

Therefore, addressing these two key issues—ensuring weight boundedness and satisfying input constraints—is essential for the reliable design of neuro-adaptive controllers. The following section will provide a detailed review of the existing solutions to these challenges.

B. Literature Review

1) Ensuring Weight Norm Boundedness: A common challenge in neuro-adaptive control is maintaining the boundedness of the NN weights to ensure stability. In many studies, projection operators were employed to enforce upper bounds on the weight norms, ensuring that the weights do not grow unboundedly. For example, in [8], [13], [16], projection operators were used to constrain the weight norms to remain below predetermined constants. However, these constants were often selected as large as possible due to the lack of theoretical guarantees regarding the global optimality of the weight val-

Manuscript created Month, 2024; This work was supported by Korea Research Institute for Defense Technology planning and advancement (KRIT) grant funded by Korea government DAPA (Defense Acquisition Program Administration) (No. KRIT-CT-22-087, Synthetic Battlefield Environment based Integrated Combat Training Platform). *(Corresponding author: Kyunghwan Choi)*

Myeongseok Ryu, Donghwan Hong, and Kyunghwan Choi are with the School of Mechanical and Robotics Engineering, Gwangju Institute of Science and Technology, Gwangju 61005, Republic of Korea (e-mail: khchoi@gist.ac.kr).

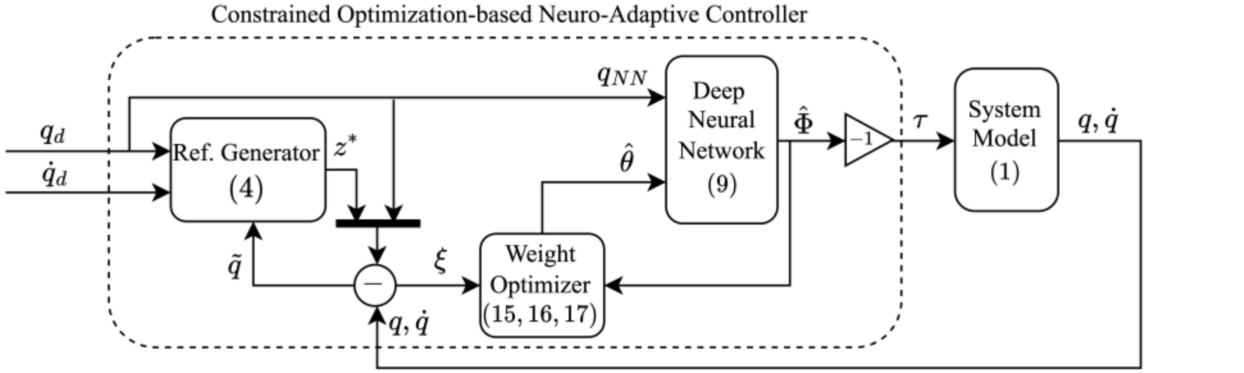


Fig. 1: Architecture of the constrained optimization-based neuro-adaptive controller (CoNAC).

Constrained optimization problem

$$\min_{\hat{\theta}} \ J(\xi; \hat{\theta})$$

$$\text{s.t. } c_j(\hat{\theta}) \leq 0, \quad j \in \mathcal{I},$$

Update laws to solve the problem

$$\dot{\hat{\theta}} = -\alpha \frac{\partial L}{\partial \hat{\theta}} = -\alpha \left(\frac{\partial J}{\partial \hat{\theta}} + \sum_{j \in \mathcal{A}} \lambda_j \frac{\partial c_j}{\partial \hat{\theta}} \right),$$

$$\dot{\lambda}_j = \beta_j \frac{\partial L}{\partial \lambda_j} = \beta_j c_j, \qquad \forall j \in \mathcal{A},$$

$$\lambda_j = \max(\lambda_j, 0), \qquad \forall j \in \mathcal{A},$$

Lyapunov-based stability guarantee

Theorem 1. *For the dynamical system in (1), the neuro-adaptive controller (9) and weight adaptation laws (15), (16), and (17) ensure the boundedness of the augmented error ξ and the weight estimate $\hat{\theta}$. This holds with the weight norm constraint (21) and input constraints satisfying Assumption 1 and 2, provided that the control gains k_q and k_z satisfy (32).*

Extension of Lyapunov-based approach

- **2. Physics-Informed Online Learning of Flux Linkage Model for Synchronous Machines (In p**

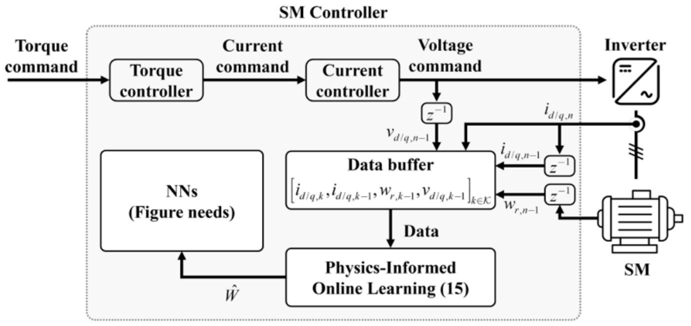
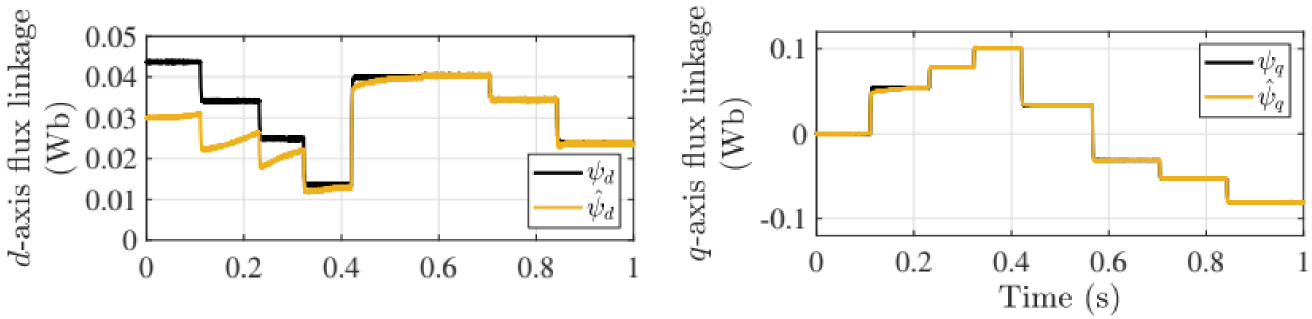


Fig. 1. Schematic diagram of the proposed physics-informed online learning method for the flux linkage model of SMs.



Constrained optimization problem

$$\min_{\hat{W}} J(\hat{W}) = w_p J_p(\hat{W}) + w_d J_d(\hat{W})$$

subject to

$$c^{eq}(\hat{W}) = \hat{\psi}_q(\hat{W}_q, 0, 0) = 0,$$
$$c_1^{in}(\hat{W}) = \hat{\psi}_d(\hat{W}_d, 0, 0) \geq \underline{\lambda}_{pm},$$
$$c_2^{in}(\hat{W}) = \hat{L}_{dd}(\hat{W}_d, i_d, i_q) \geq \underline{L}_{dd},$$
$$c_3^{in}(\hat{W}) = \hat{L}_{qq}(\hat{W}_q, i_d, i_q) \geq \underline{L}_{qq},$$

for $(i_d, i_q) = (), (), \ldots$

Update laws to solve the problem

$$\dot{\hat{W}} = -\alpha \frac{\partial L(\hat{W}, \lambda^{eq}, \lambda^{in})}{\partial \hat{W}},$$
$$\dot{\lambda}^{eq} = \beta^{eq} c^{eq}(\hat{W}),$$
$$\dot{\lambda}_j^{in} = \beta_j^{in} c_j^{in}(\hat{W}), \forall j \in \mathcal{A},$$

Lyapunov-based convergence guarantee(?)

Theorem 1 *Suppose the sampling time T_s is sufficiently small, and c_j^{in} is instantly active during the online learning process. Then, the learning rules (15) update the estimated weight vector $\hat{W}$, as well as the Lagrangian multipliers λ^{eq} and λ_j^{in}, to satisfy the first-order optimality conditions for the loss function $L(\hat{W}, \lambda^{eq}, \lambda^{in})$, achieving quadratic convergence.*

Extension of Lyapunov-based approach

- **2. Physics-Informed Online Learning of Flux Linkage Model for Synchronous Machines (In preparation)**
  - Use of protot ons?

IEEE Open Journal of the Industrial Electronics Society

Received 26 February 2022; revised 17 March 2022; accepted 22 March 2022. Date of publication 25 March 2022; date of current version 12 May 2022. The review of this paper was arranged by Associate Editor Haitham Abu-Rub.

Digital Object Identifier 10.1109/OJIES.2022.3162336

Analytical Prototype Functions for Flux Linkage Approximation in Synchronous Machines

SHIH-WEI SU [1], CHRISTOPH M. HACKL [2] (Senior Member, IEEE), AND RALPH KENNEL [1] (Senior Member, IEEE)

[1] Institute of Electrical Drive System and Power Electronics, Technical University of Munich, 80333 Munich, Germany
[2] Department of Electrical Engineering and Information Technology, Hochschule München University of Applied Sciences, 80335 Munich, Germany

CORRESPONDING AUTHOR: SHIH-WEI SU (e-mail: shihwei.su@tum.de).

ABSTRACT Physically motivated and analytical prototype functions are proposed to approximate the nonlinear flux linkages of nonlinear synchronous machines (SMs) in general; and reluctance synchronous machines (RSMs) and interior permanent magnet synchronous machines (IPMSMs) in particular. Such analytical functions obviate the need of huge lookup tables (LUTs) and are beneficial for optimal operation management and nonlinear control of such machines. The proposed flux linkage prototype functions are capable of mimicking the nonlinear self-axis and cross-coupling saturation effects of SMs. Moreover, the differentiable prototype functions allow to easily derive analytical expressions for the differential inductances by simple differentiation of the analytical flux linkage prototype functions. In total, two types of flux linkage prototype functions are developed. The first flux linkage approximation is rather simple and obeys the energy conservation rule for "symmetric" flux linkages of RSMs. With the gained knowledge, the second type of prototype functions is derived in order to achieve approximation flexibility necessary for SMs with permanent (or electrical) excitation with "unsymmetric" flux linkages due to the excitation offset. All proposed flux linkage prototype functions are continuously differentiable, obey the energy conservation rule and, as fitting results show, achieve a (very) high approximation accuracy over the whole operation range.

INDEX TERMS Analytical flux linkage prototype functions, interior permanent magnet synchronous machine, reluctance synchronous machine, saturation effects.

I. INTRODUCTION

With the developed manufacturing and control techniques and the increased efficiency requirements, induction machines (IMs) are more and more replaced by synchronous machines (SMs) [1]. Reluctance synchronous machines (RSMs) and permanent magnet synchronous machines (PMSMs) achieve higher efficiencies and better overall performance.

Except surface-mounted PMSM (SPMSM), both RSM and interior PMSM (IPMSM) exhibit significant magnetic saturation [2], [3], resulting in highly nonlinear flux linkages which depend on not only the direct-axis current but also the quadrature-axis current, leading to magnetic cross-coupling. In order to achieve the best possible drive performance, the saturation and cross-coupling effects cannot be neglected. The effectiveness of developed control algorithms, e.g., nonlinear current control strategies [4], optimal feed-forward torque control (OFTC) [5], [6] or model predictive control [7], is deteriorated by model and parameter uncertainties. In addition, cross-coupling inductances [8] lead to position estimation errors in encoderless control. Consequently, a comprehensive flux linkage (or differential inductance) model is essential for the control of modern electrical drive systems.

The magnetic nonlinearity of the flux linkage maps can normally be extracted (mostly as LUTs) by using finite element analysis (FEA) or by conducting experiments in the laboratory. For many application (e.g. industrial drives), FEA data from the machine manufacturers may not be available to commissioning or control engineers but—as it is required for optimal controller tuning and operation of the

$$\widehat{\psi}_s^d(i_s^d, i_s^q) = \widehat{\psi}_{s,\mathrm{self}}^d(i_s^d) \underbrace{- \frac{1}{F(I_{d1})G(I_{q1})} \Big(G(I_{q1})F'(i_s^d)\Big)\Big(F(I_{d1})G(i_s^q)\Big)}_{\overset{(10)}{=}\widehat{\psi}_{s,\mathrm{cross}}^d(i_s^d, i_s^q)} \tag{23}$$

and

$$\widehat{\psi}_s^q(i_s^d, i_s^q) = \widehat{\psi}_{s,\mathrm{self}}^q(i_s^q) \underbrace{- \frac{1}{F(I_{d1})G(I_{q1})} \Big(G(I_{q1})F(i_s^d)\Big)\Big(F(I_{d1})G'(i_s^q)\Big)}_{\overset{(11)}{=}\widehat{\psi}_{s,\mathrm{cross}}^q(i_s^d, i_s^q)}. \tag{24}$$

Visiting of Myeongseok

- **Myeongseok's publications**

[1] M. Ryu, D. Hong, and K. Choi*, "Constrained Optimization-Based Neuro-Adaptive Control (CoNAC) for Uncertain Euler-Lagrange Systems Under Weight and Input Constraints," **Under review for *IEEE Transactions on Neural Networks and Learning Systems***.

[2] M. Ryu, J. Kim, and K. Choi*, "Imposing Weight Norm Constraint for Neuro-Adaptive Control," **Under review for *European Control Conferences*.**

[3] M. Ryu and K. Choi*, "CNN-based End-to-End Adaptive Controller with Stability Guarantees," *arXiv*, Preprint, arXiv.2403.03499.

[4] M. Ryu, J. Ha, M. Kim, and K. Choi*, "A Comparative Study of Reinforcement Learning and Analytical Methods for Optimal Control," in ***2023 International Workshop on Intelligent Systems (IWIS)***, 2023: IEEE

[3] S. Jang, M. Ryu, and K. Choi*, "Physics-Informed Online Learning of Flux Linkage Model for Synchronous Machines," **In preparation**.