Practical Techniques for Reachability-Based Safety Verification of Real-World Neural Feedback Loops

Nicholas Rober
Ph.D. Candidate

Ph.D. Thesis Proposal

Department of Aeronautics and Astronautics
Massachusetts Institute of Technology

Thesis Committee:

Professor Jonathan P. How (Chair)
Professor Chuchu Fan
Professor Mykel J. Kochenderfer

External Evaluator:

Professor Esen Yel

August 13, 2025

Abstract

Neural networks (NNs) are becoming increasingly popular in the design of control pipelines for autonomous systems. However, due to the unpredictable nature of NNs, systems that have NNs in their control pipelines, i.e., neural feedback loops (NFLs), need safety assurances before they can be applied in safety-critical situations. To address this need, this thesis investigates reachability for safety verification of practical NFL verification problems. In practice, NFLs often have sophisticated autonomy pipelines consisting of estimation, planning, and control modules with highly nonlinear closed-loop behavior that needs to be verified over long time horizons. These factors pose challenges to existing techniques, which calculate reachable set over-approximations (RSOAs) as a tractable alternative to exact reachable sets. While relatively efficient to calculate, RSOAs can be prohibitively conservative given nonlinear dynamics and NN control policies. Moreover, existing techniques consider simple control pipelines, typically consisting of just an NN controller with full state feedback. To address these issues, this thesis proposes three contributions: First, we propose a method to reduce conservativeness of RSOAs for NFLs in a way that scales well with both the state dimension of the system and the time horizon of the verification problem. Second, we propose methods to incorporate reachability into the NN controller's learning pipeline to create verifiable-by-design policies that can be more easily verified as safe. Finally, we investigate ways to verify safety for NFLs that consist of complicated autonomy pipelines consisting of estimation, planning, and control modules, any combination of which may be NN-based. The results of this thesis will greatly broaden the types of NFL verification problems that can be solved, thus enabling the safe application of NFLs for real-world autonomy.

1 Problem Statement

This thesis investigates the problem of safety verification for autonomous systems that have neural networks (NNs) embedded in feedback loops within their autonomy pipelines. These systems are hereafter referred to as neural feedback loops (NFLs) and pose specific challenges with respect to safety verification due to the complicated and nonlinear nature of the underlying NNs. Interval-based reachability analysis provides a highly generalizeable framework for safety verification of NFLs, but needs further advancement to address verification problems representative of the real-world application of NFLs. In the following section, we elaborate on three key challenges that must be addressed to solve real-world safety verification problems for NFLs, along with proposed solutions to these challenges.

1.1 Scale of NFL Verification Problems

To solve practical safety verification problems for NFLs, a reachability tool must be able to efficiently generate reachable sets over long time horizons while scaling well with respect to the system's state dimension. This is challenging because calculating exact reachable sets is computationally expensive, and cheaper reachable set over-approximations (RSOAs) suffer from the *wrapping effect* [32], causing them to become excessively conservative over long time horizons. The conservativeness of RSOAs poses a challenge to safety verification because overly conservative RSOAs can indicate a violation of safety constraints even if the system is safe. To combat the wrapping effect, refinement techniques are typically employed to reduce the conservativeness of RSOAs, but they introduce additional computational challenges.

Partitioning [16,20,44,63] accomplishes refinement by splitting up the initial state set and calculating reachable sets for each of the resulting subsets. This allows for tighter relaxations of the NN and thus less conservative RSOAs. While partitioning is an effective approach for some problems, splitting up the initial set is a strategy that scales poorly with the state dimension of the system being controlled.

Another approach to refinement lies in symbolic reachability calculations [9]. Symbolic RSOA calculations generate bounds on states $N > 1$ time steps in the future, thus mitigating the wrapping effect. However, since an N-step calculation involves analyzing N closed-loop time steps, symbolic calculations are very difficult for long time horizons. Sidrane et al. [49] overcame this challenge by alternating between symbolic and one-step concrete calculations on a predefined schedule, but it was not clear how this schedule should be determined. Recent work [51] addresses the scheduling question by determining a hybrid-symbolic schedule given a specified time budget, but does not consider how the resulting RSOAs interact with safety constraints on the NFL, and thus may not refine the RSOAs in a way that verifies safety.

Thus, concrete and symbolic RSOA calculations each have their tradeoffs: concrete calculations are fast, but suffer from being overly conservative over multiple time steps whereas symbolic calculations are slower over long time horizons, but are much less conservative. Given these tradeoffs, it is unknown how to efficiently calculate RSOAs to verify if the state of a high-dimensional NFL stays in the safe region of the state space over a long time horizon.

1.2 Learning Verifiably Safe Control Policies

Even with a state-of-the-art refinement approach, policies that operate near the border of an unsafe region may still be very difficult to verify. Standard training methods find an optimal policy given cost function, but do not consider that they need to be verified as safe. Training in this manner leads to policies that perform very well, but which may not be *verifiable*, as shown in Fig. 1. The policy in Fig. 1 was trained using standard imitation learning (IL) using mean squared error loss with state-action pairs generated by an MPC controller that drives the state from the black box to the origin. Notice that the simulated trajectories pass very close to the constraint $\mathbf{x}_2 \geq -1$ (gray), which could introduce a significant challenge during the verification process. Simultaneously, arbitrarily inflating the constraint could lead to an overly conservative policy that performs worse than the original.

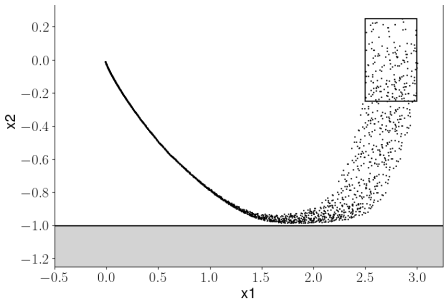

Figure 1: Standard IL policy operates very close to constraint.

To address this challenge, several recent works have investigated Verification-in-the-loop learning (VILL), but this adds an additional computational cost which must be reduced. Recent work [62] used VILL in a reinforcement learning context and started to address the challenge of reducing computational cost, but is still considerably slower than standard learning pipelines Alternatively, [60] uses reach-avoid properties with an approximate gradient descent method, but does not enable other learning objectives and is thus overly restrictive. Another line of work [37, 56] imposes constraints on the learned policy, which can be used to ensure safety, but is limited in the number of convex constraints that can be applied and requires an iterative approach on the forward call of the NN [37] or cannot apply system-level constraints [56]. Thus, the problem of efficiently incorporating verification in the training loop to create policies that are verifiably safe with respect to general nonconvex constraints is still an open problem that must be addressed.

1.3 Verifying Complicated Autonomy Pipelines

Real-world NFLs are rarely a simple NN controller giving an input to the system. NNs may be integrated at many points in a systems autonomy pipeline, including perception/estimation modules [42, 66], planners [29, 55], and low-level control [48]. Consider the NFLs with various combinations of learning-enabled and standard components shown in Fig. 2.

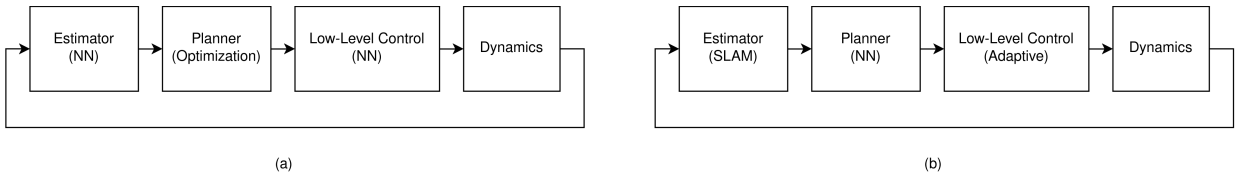

Figure 2: NFLs with various combinations of learning-enabled and standard components.

While recent works have developed methods to verify systems with NN estimators and controllers [27, 41], the presence of an optimization-based planner in loop (a) introduces complexity that cannot be handled with existing methods. Moreover, while the only NN component in (b) is the planner, existing methods for verification of NFLs do not consider ways to incorporate standard components like SLAM-based localization or common control techniques, such as adaptive control. Since Fig. 2 is representative of common implementations of NFLs, there is a great need to develop methods capable of verifying their safety.

2 Proposed Contributions

2.1 Scalable Techniques for Safety Verification of NFLs

In response to the problem outlined in Section 1.1, this thesis proposes the following contributions: 1) Constraint-Aware Refinement for Verification (CARV): a framework that explicitly uses the system's constraints to guide the safety verification process for NFLs; 2) A refinement algorithm that finds a hybrid-symbolic schedule to enable efficient safety verification for NFLs while avoiding expensive RSOA calculations and still mitigating the wrapping effect; and 3) Experiments wherein CARV verifies safety for a problem where other approaches either fail or take more than 60× longer and require 40× more memory.

The key insight of CARV is that refinement is only conducted as needed to efficiently verify safety for a given problem. Moreover, by using symbolic calculations but setting a limit on the length of any given symbolic step, CARV scales well with both state dimension and time horizon. These advancements allow CARV to successfully and efficiently verify safety for a 6D nonlinear quadrotor, shown in Fig. 3 as well as a 3D ground robot model and a 2D double integrator, as shown in Table 1. More details on the implementation of CARV can be found in [45].

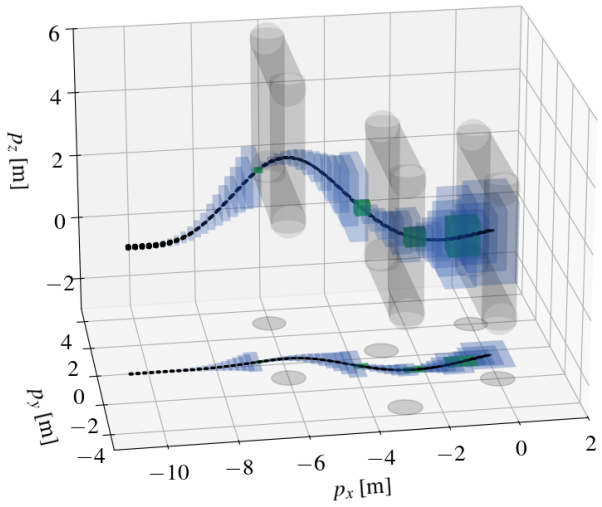

Figure 3: CARV verifies safety for a 6D quadrotor.

Table 1: CARV efficiently verifies safety for each problem.

Approach	DI		GR		QD	
	Time	**Verif**	**Time**	**Verif**	**Time**	**Verif**
part [16]	20.79 s	Y	540.10 s	Y	—	N
symb [9]	29.42 s	Y	—	N	—	N
unif [49]	1.77 s	N	9.71 s	N	35.41 s	N
CARV (**ours**)	3.11 s	Y	9.32 s	Y	32.48 s	Y

2.2 Efficient Verification-in-the-Loop Learning

To address the problem posed by Section 1.2, this thesis will develop an approach that uses VILL to efficiently learn control policies that are verifiably safe with respect to nonconvex constraints. We will use insights from previous work on backward reachability [43, 46] for nonconvex problems and efficient verification [45] to help develop this approach. Moreover, we note that since post-learning verification is still necessary, the verification used in the learning process does not need to be sound. With this in mind, we have started development of a sim-guided reachability approach that does not provide formal guarantees, but provides approximate reachable sets with an order-of-magnitude speedup over similar sound methods. Fig. 4 shows an early implementation of this approach where the blue sets show the results of our pseudo reachabilty (blue), calculated in 0.33 s, compared with a symbolic reachability calculation (green), calculated in 31.80 s. Though the pseudo reachability approach is not guaranteed to contain all trajectories, it is visually similar and was calculated in much less time.

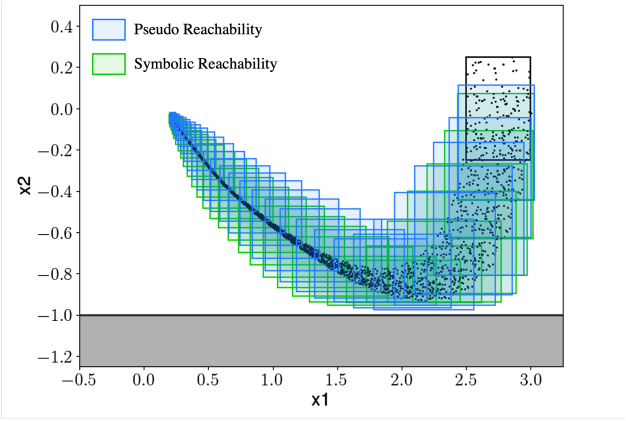

Figure 4: Symbolic (sound) reachability vs pseudo reachability.

With further development of this work, this thesis proposes the following contributions: 1) Verification-in-the-loop learning with sim-guided pseudo reachability to create verifiably safe control policies. 2) Investigation of sampling approaches to efficiently approximate reachable sets for use in the pseudo reachability calculation. 3) Extension of concepts from [45] to minimize computational impact of verification on the learning process. 4) Integration of contributions (1-3) with backward reachability-based verification approach [46] to handle nonconvex constraints.

2.3 Verifying NFLs with Complicated Autonomy Pipelines

To address the problem posed in Section 1.3 this thesis will develop an approach to verify safety for NFLs with various combinations of learning-enabled and standard components in their feedback loops. We will take inspiration from [64], which, in the context of open-loop neural verification, extended standard neural verification tools to more general computational graphs to enable verification of a wide variety of neural architectures. Similarly, we will take inspiration from [26], which connected neural verification to the wide body of work on hybrid automota by transforming the network into a hybrid system. The goal of this work will be to combine guarantees from neural verification with those associated with standard approaches, e.g., guaranteed safe trajectory planning using optimization, to ensure safety for a system that uses each of these components. The main challenge will be combining these guarantees under a unified framework that can be analyzed using and/or extending existing tools for safety verification.

3 Proposed Schedule

Date	Milestone
May 2023	Passed doctoral field evaluation.
October 2024	First committee meeting.
December 2024	Acceptance of "Constraint-Aware Refinement for Safety Verification of Neural Feedback Loops" to L-CSS [45], detailed in Section 2.1.
December 2024	**Thesis proposal defense**.
Spring 2025	Efficient verification-in-loop learning (to be submitted to CDC'25), detailed in Section 2.2 and initial work on journal extension incorporating sampling strategy and backward-reachability for nonconvex problems.
Summer 2025	Submission of journal extension of CDC'25 paper (to be submitted to TAC or OJ-CSYS). Ideation and initial work on verification of complicated autonomy pipelines, detailed in Section 2.3.
Fall 2025	Submission of verification for complicated autonomy pipelines (to be submitted to RA-L). Thesis writing and iteration.
May 2026	Thesis defense.

4 Work to Date

Main publications:

- Constraint-Aware Refinement for Safety Verification of Neural Feedback Loops [45] (accepted for publication in L-CSS)
- Backward Reachability Analysis for Neural Feedback Loops: Techniques for Linear and Nonlinear Systems [46] (OJ-CSYS)
- Backward Reachability Analysis for Neural Feedback Loops [43] (CDC '22)

Other publications:

- Safe Autonomy for Uncrewed Surface Vehicles Using Adaptive Control and Reachability Analysis [36] (submitted to TCST)
- Online Data-Driven Safety Certification for Systems Subject to Unknown Disturbances [47] (ICRA '24)
- A Hybrid Partitioning Strategy for Backward Reachability Analysis of Neural Feedback Loops [44] (ACC '23)

References

[1] Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *International conference on machine learning*, pages 22–31. PMLR, 2017.

[2] Anil Alan, Andrew J Taylor, Chaozhe R He, Aaron D Ames, and Gábor Orosz. Control barrier functions and input-to-state safety with application to automated vehicles. *IEEE Transactions on Control Systems Technology*, 31(6):2744–2759, 2023.

[3] Aaron D Ames, Samuel Coogan, Magnus Egerstedt, Gennaro Notomista, Koushil Sreenath, and Paulo Tabuada. Control barrier functions: Theory and applications. In *2019 18th European control conference (ECC)*, pages 3420–3431. IEEE, 2019.

[4] Somil Bansal, Mo Chen, Sylvia Herbert, and Claire J Tomlin. Hamilton-Jacobi reachability: A brief overview and recent advances. In *IEEE Conference on Decision and Control (CDC)*, pages 2242–2253, 2017.

[5] Somil Bansal and Claire J Tomlin. Deepreach: A deep learning approach to high-dimensional reachability. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1817–1824. IEEE, 2021.

[6] Christopher Brix, Stanley Bak, Changliu Liu, and Taylor T Johnson. The fourth international verification of neural networks competition (vnn-comp 2023): Summary and results. *arXiv preprint arXiv:2312.16760*, 2023.

[7] Christopher Brix, Mark Niklas Müller, Stanley Bak, Taylor T Johnson, and Changliu Liu. First three years of the international verification of neural networks competition (vnn-comp). *International Journal on Software Tools for Technology Transfer*, 25(3):329–339, 2023.

[8] Hao Chen, Gonzalo E Constante Flores, and Can Li. Physics-informed neural networks with hard linear equality constraints. *Computers & Chemical Engineering*, 189:108764, 2024.

[9] Shaoru Chen, Victor M Preciado, and Mahyar Fazlyab. One-shot reachability analysis of neural network dynamical systems. In *Int. Conference on Robotics and Automation (ICRA)*, pages 10546–10552. IEEE, 2023.

[10] Charles Dawson, Sicun Gao, and Chuchu Fan. Safe control with learned certificates: A survey of neural Lyapunov, barrier, and contraction methods. *arXiv preprint arXiv:2202.11762*, 2022.

[11] DeepMind. jax_verify, 2020.

[12] Zihao Dong, Shayegan Omidshafiei, and Michael Everett. Collision avoidance verification of multiagent systems with learned policies. *IEEE Control Systems Letters*, 2024.

[13] Hai Duong, ThanhVu Nguyen, and Matthew Dwyer. A dpll (t) framework for verifying deep neural networks. *arXiv preprint arXiv:2307.10266*, 2023.

[14] Taha Entesari, Sina Sharifi, and Mahyar Fazlyab. Reachlipbnb: A branch-and-bound method for reachability analysis of neural autonomous systems using lipschitz bounds. *arXiv preprint arXiv:2211.00608*, 2022.

[15] Lawrence C Evans. Graduate studies in mathematics, 1998.

[16] Michael Everett, Golnaz Habibi, Chuangchuang Sun, and Jonathan P How. Reachability analysis of neural feedback loops. *IEEE Access*, 9:163938–163953, 2021.

[17] Jiameng Fan, Chao Huang, Xin Chen, Wenchao Li, and Qi Zhu. Reachnn*: A tool for reachability analysis of neural-network controlled systems. In *International Symposium on Automated Technology for Verification and Analysis*, pages 537–542, 2020.

[18] Yuang Geng, Jake Brandon Baldauf, Souradeep Dutta, Chao Huang, and Ivan Ruchkin. Bridging dimensions: Confident reachability for high-dimensional controllers. In *International Symposium on Formal Methods*, pages 381–402. Springer, 2024.

[19] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[20] Akash Harapanahalli, Saber Jafarpour, and Samuel Coogan. Contraction-guided adaptive partitioning for reachability analysis of neural network controlled systems. In *2023 62nd IEEE Conference on Decision and Control (CDC)*, pages 6044–6051. IEEE, 2023.

[21] Sylvia Herbert, Jason J Choi, Suvansh Sanjeev, Marsalis Gibson, Koushil Sreenath, and Claire J Tomlin. Scalable learning of safety guarantees for autonomous systems using hamilton-jacobi reachability. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5914–5920. IEEE, 2021.

[22] Kerianne L Hobbs, Mark L Mote, Matthew CL Abate, Samuel D Coogan, and Eric M Feron. Runtime assurance for safety-critical systems: An introduction to safety filtering approaches for complex control systems. *IEEE Control Systems Magazine*, 43(2):28–65, 2023.

[23] Haimin Hu, Mahyar Fazlyab, Manfred Morari, and George J Pappas. Reach-SDP: Reachability analysis of closed-loop systems with neural network controllers via semidefinite programming. In *IEEE Conference on Decision and Control (CDC)*, pages 5929–5934, 2020.

[24] Yanlong Huang and Darwin G Caldwell. A linearly constrained nonparametric framework for imitation learning. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4400–4406. IEEE, 2020.

[25] Radoslav Ivanov, Taylor Carpenter, James Weimer, Rajeev Alur, George Pappas, and Insup Lee. Verisig 2.0: Verification of neural network controllers using taylor model preconditioning. In *International Conference on Computer Aided Verification*, pages 249–262. Springer, 2021.

[26] Radoslav Ivanov, James Weimer, Rajeev Alur, George J Pappas, and Insup Lee. Verisig: verifying safety properties of hybrid systems with neural network controllers. In *International Conference on Hybrid Systems: Computation and Control*, pages 169–178, 2019.

[27] Sydney M Katz, Anthony L Corso, Christopher A Strong, and Mykel J Kochenderfer. Verification of image-based neural network controllers using generative models. *Journal of Aerospace Information Systems*, 19(9):574–584, 2022.

[28] Niklas Kochdumper, Hanna Krasowski, Xiao Wang, Stanley Bak, and Matthias Althoff. Provably safe reinforcement learning via action projection using reachability analysis and polynomial zonotopes. *IEEE Open Journal of Control Systems*, 2:79–92, 2023.

[29] Kota Kondo, Claudius T Tewari, Andrea Tagliabue, Jesus Tordesillas, Parker C Lusk, and Jonathan P How. Primer: Perception-aware robust learning-based multiagent trajectory planner. *arXiv preprint arXiv:2406.10060*, 2024.

[30] Alexei Kopylov, Stefan Mitsch, Aleksey Nogin, and Michael Warren. Formally verified safety net for waypoint navigation neural network controllers. In *Formal Methods: 24th International Symposium, FM 2021, Virtual Event, November 20–26, 2021, Proceedings 24*, pages 122–141. Springer, 2021.

[31] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial examples in the physical world. In *International Conference on Learning Representations (ICLR) Workshop Track Proceedings*, 2017.

[32] Colas Le Guernic. *Reachability analysis of hybrid systems with linear continuous dynamics*. PhD thesis, Université Joseph-Fourier-Grenoble I, 2009.

[33] Lars Lindemann, Yiqi Zhao, Xinyi Yu, George J Pappas, and Jyotirmoy V Deshmukh. Formal verification and control with conformal prediction. *arXiv preprint arXiv:2409.00536*, 2024.

[34] Diego Manzanas Lopez, Sung Woo Choi, Hoang-Dung Tran, and Taylor T Johnson. Nnv 2.0: the neural network verification tool. In *International Conference on Computer Aided Verification*, pages 397–412. Springer, 2023.

[35] Mayra Macas, Chunming Wu, and Walter Fuertes. Adversarial examples: A survey of attacks and defenses in deep learning-enabled cybersecurity systems. *Expert Systems with Applications*, 238:122223, 2024.

[36] Karan Mahesh, Tyler M Paine, Max L Greene, Nicholas Rober, Steven Lee, Sildomar T Monteiro, Anuradha Annaswamy, Michael R Benjamin, and Jonathan P How. Safe autonomy for uncrewed surface vehicles using adaptive control and reachability analysis. *arXiv preprint arXiv:2410.01038*, 2024.

[37] Youngjae Min, Anoopkumar Sonar, and Navid Azizan. Hard-constrained neural networks with universal approximation guarantees. *arXiv preprint arXiv:2410.10807*, 2024.

[38] Ian M Mitchell, Alexandre M Bayen, and Claire J Tomlin. A time-dependent hamilton-jacobi formulation of reachable sets for continuous dynamic games. *IEEE Transactions on automatic control*, 50(7):947–957, 2005.

[39] Arnold Neumaier. *The wrapping effect, ellipsoid arithmetic, stability and confidence regions*. Springer, 1993.

[40] Charles Noren, Weiye Zhao, and Changliu Liu. Safe adaptation with multiplicative uncertainties using robust safe set algorithm. *IFAC-PapersOnLine*, 54(20):360–365, 2021.

[41] Corina S Păsăreanu, Ravi Mangal, Divya Gopinath, Sinem Getir Yaman, Calum Imrie, Radu Calinescu, and Huafeng Yu. Closed-loop analysis of vision-based autonomous systems: A case study. In *International conference on computer aided verification*, pages 289–303. Springer, 2023.

[42] Cristiano Premebida, Rares Ambrus, and Zoltan-Csaba Marton. Intelligent robotic perception systems. *Applications of Mobile Robots*, pages 111–127, 2018.

[43] Nicholas Rober, Michael Everett, and Jonathan P How. Backward reachability analysis for neural feedback loops. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, pages 2897–2904. IEEE, 2022.

[44] Nicholas Rober, Michael Everett, Songan Zhang, and Jonathan P How. A hybrid partitioning strategy for backward reachability of neural feedback loops. In *2023 American Control Conference (ACC)*, pages 3523–3528. IEEE, 2023.

[45] Nicholas Rober and Jonathan P How. Constraint-aware refinement for safety verification of neural feedback loops. *arXiv preprint arXiv:2410.00145*, 2024.

[46] Nicholas Rober, Sydney M Katz, Chelsea Sidrane, Esen Yel, Michael Everett, Mykel J Kochenderfer, and Jonathan P How. Backward reachability analysis of neural feedback loops: Techniques for linear and nonlinear systems. *IEEE Open Journal of Control Systems*, 2:108–124, 2023.

[47] Nicholas Rober, Karan Mahesh, Tyler M Paine, Max L Greene, Steven Lee, Sildomar T Monteiro, Michael R Benjamin, and Jonathan P How. Online data-driven safety certification for systems subject to unknown disturbances. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9939–9945. IEEE, 2024.

[48] Guanya Shi, Xichen Shi, Michael O'Connell, Rose Yu, Kamyar Azizzadenesheli, Animashree Anandkumar, Yisong Yue, and Soon-Jo Chung. Neural lander: Stable drone landing control using learned dynamics. In *2019 international conference on robotics and automation (icra)*, pages 9784–9790. IEEE, 2019.

[49] Chelsea Sidrane, Amir Maleki, Ahmed Irfan, and Mykel J Kochenderfer. Overt: An algorithm for safety verification of neural network control policies for nonlinear systems. *Journal of Machine Learning Research*, 23(117):1–45, 2022.

[50] Chelsea Sidrane, Amir Maleki, Ahmed Irfan, and Mykel J Kochenderfer. OVERT: An algorithm for safety verification of neural network control policies for nonlinear systems. *Journal of Machine Learning Research*, 23(117):1–45, 2022.

[51] Chelsea Sidrane and Jana Tumova. TTT: A temporal refinement heuristic for tenuously tractable discrete time reachability problems. *arXiv preprint arXiv:2407.14394*, 2024.

[52] Oswin So, Zachary Serlin, Makai Mann, Jake Gonzales, Kwesi Rutledge, Nicholas Roy, and Chuchu Fan. How to train your neural control barrier function: Learning safety filters for complex input-constrained systems. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11532–11539. IEEE, 2024.

[53] Adam Stooke, Joshua Achiam, and Pieter Abbeel. Responsive safety in reinforcement learning by pid lagrangian methods. In *International Conference on Machine Learning*, pages 9133–9143. PMLR, 2020.

[54] Andrea Tagliabue, Dong-Ki Kim, Michael Everett, and Jonathan P How. Efficient guided policy search via imitation of robust tube mpc. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 462–468. IEEE, 2022.

[55] Jesus Tordesillas and Jonathan P How. Deep-panther: Learning-based perception-aware trajectory planner in dynamic environments. *IEEE Robotics and Automation Letters*, 8(3):1399–1406, 2023.

[56] Jesus Tordesillas, Jonathan P How, and Marco Hutter. Rayen: Imposition of hard convex constraints on neural networks. *arXiv preprint arXiv:2307.08336*, 2023.

[57] Hoang-Dung Tran, Neelanjana Pal, Diego Manzanas Lopez, Patrick Musau, Xiaodong Yang, Luan Viet Nguyen, Weiming Xiang, Stanley Bak, and Taylor T Johnson. Verification of piecewise deep neural networks: a star set approach with zonotope pre-filter. *Formal Aspects of Computing*, 33:519–545, 2021.

[58] Joseph A Vincent and Mac Schwager. Reachable polyhedral marching (RPM): A safety verification algorithm for robotic systems with deep neural network components. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 9029–9035, 2021.

[59] Kim P Wabersich, Andrew J Taylor, Jason J Choi, Koushil Sreenath, Claire J Tomlin, Aaron D Ames, and Melanie N Zeilinger. Data-driven safety filters: Hamilton-jacobi reachability, control barrier functions, and predictive methods for uncertain systems. *IEEE Control Systems Magazine*, 43(5):137–177, 2023.

[60] Yixuan Wang, Chao Huang, Zhaoran Wang, Zhilu Wang, and Qi Zhu. Verification in the loop: Correct-by-construction control learning with reach-avoid guarantees. *arXiv preprint arXiv:2106.03245*, 2021.

[61] Tianhao Wei, Shucheng Kang, Weiye Zhao, and Changliu Liu. Persistently feasible robust safe control by safety index synthesis and convex semi-infinite programming. *IEEE Control Systems Letters*, 7:1213–1218, 2022.

[62] Junlin Wu, Huan Zhang, and Yevgeniy Vorobeychik. Verified safe reinforcement learning for neural network dynamic models. *arXiv preprint arXiv:2405.15994*, 2024.

[63] Weiming Xiang, Hoang-Dung Tran, and Taylor T Johnson. Specification-guided safety verification for feedforward neural networks. *arXiv preprint arXiv:1812.06161*, 2018.

[64] Kaidi Xu, Zhouxing Shi, Huan Zhang, Yihan Wang, Kai-Wei Chang, Minlie Huang, Bhavya Kailkhura, Xue Lin, and Cho-Jui Hsieh. Automatic perturbation analysis for scalable certified robustness and beyond. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:1129–1141, 2020.

[65] Huan Zhang, Tsui-Wei Weng, Pin-Yu Chen, Cho-Jui Hsieh, and Luca Daniel. Efficient neural network robustness certification with general activation functions. *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[66] Xu Zheng, Yexin Liu, Yunfan Lu, Tongyan Hua, Tianbo Pan, Weiming Zhang, Dacheng Tao, and Lin Wang. Deep learning for event-based vision: A comprehensive survey and benchmarks. *arXiv preprint arXiv:2302.08890*, 2023.

A Literature Review

A.1 Safety Verification for Neural Feedback Loops

Due to the increasing prevalence of NNs in the fields of controls and robotics, there has been a recent flurry of publications concerned with providing statements about their safety. Motivated by work demonstrating the effectiveness of adversarial attacks [19, 31, 35], many tools have been developed to analyze NNs in isolation by predicting a set of possible outputs given perturbations to a nominal input [11, 13, 34, 58, 64, 65]. VNN-COMP [6, 7] has additionally spurned the development of these tools and has introduced a set of benchmark problems, several of which shift the focus to verification of NFLs. Along with more general effort within the NFL verification community, these problems have led to a class of tools capable of generating reachable sets for NFLs [14, 16, 25, 34, 58], which will be useful for this thesis and are the subject of further discussion in Appendix A.2. Other methods for safety verification of NFLs include statistical methods based on conformal prediction [33], theorem-proving tools [30], and direct falsification [50].

Additionally, safety can be ensured at run time with the application of a safety filter [22, 59]. While not developed for NFLs in particular, safety filters correct unsafe control actions to drive a system to a safe invariant set. In contrast to the previously mentioned work on closed-loop reachability for NFLs, Hamilton-Jacobi (HJ) reachability can also be used to ensure the safety of an NFL by using the open-loop dynamics to create a safety filter [4, 15]. While HJ reachability can be used to construct high-performing safety filters, they require solving a partial differential equation using a grid of the state space, causing these methods to scale poorly with the system's state dimension [4, 38] or involve learning additional models for the purpose of reachable set calculation [5, 21]. Similarly, safety filters can be constructed using control barrier functions (CBFs) [3, 10]. Often, CBFs are hand-crafted for a specific system and safety constraint [2, 10], though this takes significant expertise and can be very difficult for complex systems. Recent work has shown promising results enabled by learning CBFs [10, 52], but it can be challenging to combat conservativeness and ensure validity of the learned safety filter, especially when input constraints are involved.

A.2 NFL Reachability and Refinement Techniques

Reachability analysis predicts the future states of a given system given uncertainty about the initial state and is the focus of many recent works [16, 17, 23, 25, 34, 49]. A big challenge many of these works face is the issue of conservativeness. While some approaches are capable of exact reachability analysis [34, 58], these are fairly limited in the range of NFLs they can be applied to, e.g., linear systems with ReLU NN controllers, and are computationally heavy. As an alternative to exact reachability calculations, there is a wide body of work on calculating Reachable Set Over Approximations (RSOAs) [9, 16, 20, 25, 49] that provide outer bounds on the exact reachable sets. Different lines of research have branched this idea out in a variety of exciting directions, including verification of perception-based controllers [18, 27, 41], multi-agent systems [12], and verification of nonconvex problems with backward reachability [43,

46]. Typically, these reachability approaches calculate RSOAs at discrete intervals and use the RSOA at time t to calculate the RSOA at time $t+1$. We refer to this approach as *concrete* RSOA calculation, which can be fast, but also accrues conservativeness due to the wrapping effect [32, 39].

There are several ways to combat the wrapping effect. First, having very tight RSOAs limits the area beyond the true reachable set, so the wrapping effect takes more time steps to manifest itself. Rich set representations, such as star sets [34, 57], can be used to help with this, but are still victim to approximation error associated with NN analysis. Alternatively, even simple set representations, such as hyper-rectangles, can be used effectively in spite of the wrapping effect via the use of refinement. Partitioning [16] splits up the initial state set and calculates reachable sets for each of the resulting subsets, thus providing tighter relaxations of the NN and an artificially enriched set representation (multiple small sets vs. one large one) leading to less conservative RSOAs. While guided partitioning strategies exist [20,44,63], these approaches ultimately suffer from the curse of dimensionality and scale poorly with the state dimension of the underlying system. Symbolic reachability [9] is another refinement approach, which can be used alone or in conjunction with partitioning. Introduced in [50], this approach calculates multiple time steps forward at once, thus allowing the RSOA at time $t+1$ to be calculated independently from the set at time t. However, symbolic calculations scale poorly with respect to the time horizon because each new time step adds more complexity to the verification problem. To address this issue, [49] proposes a hybrid-symbolic approach, alternating between symbolic andconcrete calculations on a predefined schedule. Recent work [51] extended this idea by finding a hybrid-symbolic schedule given a specified time budget.

A.3 Safe Learning

Given the potential of learned approaches for robotics applications, there has been a strong push to develop learning techniques that enable safe system behavior. Safe reinforcement learning (RL) [1, 53] has become a very popular subject of investigation, and while these approaches do reduce failure rates, they do not come with safety guarantees and still fail occasionally. To address this issue in the context of RL, [40, 61] use forward invariance and [28] uses a safety filtering approach, but these are either used in post-processing or at runtime. In the imitation learning framework (IL), work such as [54] seeks to imitate a tube MPC policy that is robust to uncertain disturbances, but again does not come with any formal guarantees. To incorporate notions of safety into the learning process, [62] used reachability calculations and a gradient approximation to create a learning pipeline based on reach-avoid properties, but is limited to a specific system and does not allow for other learning objectives. Alternatively, [62] is more generalizable since it uses some of the tools discussed in Appendix A.2, but suffers from the added computational cost.

Constrained learning is a promising direction of research that seeks to ensure safety by limiting the output of a NN. Works such as [24, 56] developed layers in a NN architecture that project the outputs onto a constraint space, thus enabling constrained NN outputs, but these works do not consider input-dependant constraints, so are not useful for verifying

safety for neural controllers. Recent work [8,37] do account for input-dependent constraints, but are limited in the number of constraints that can be applied and have no support for non-convex problems.