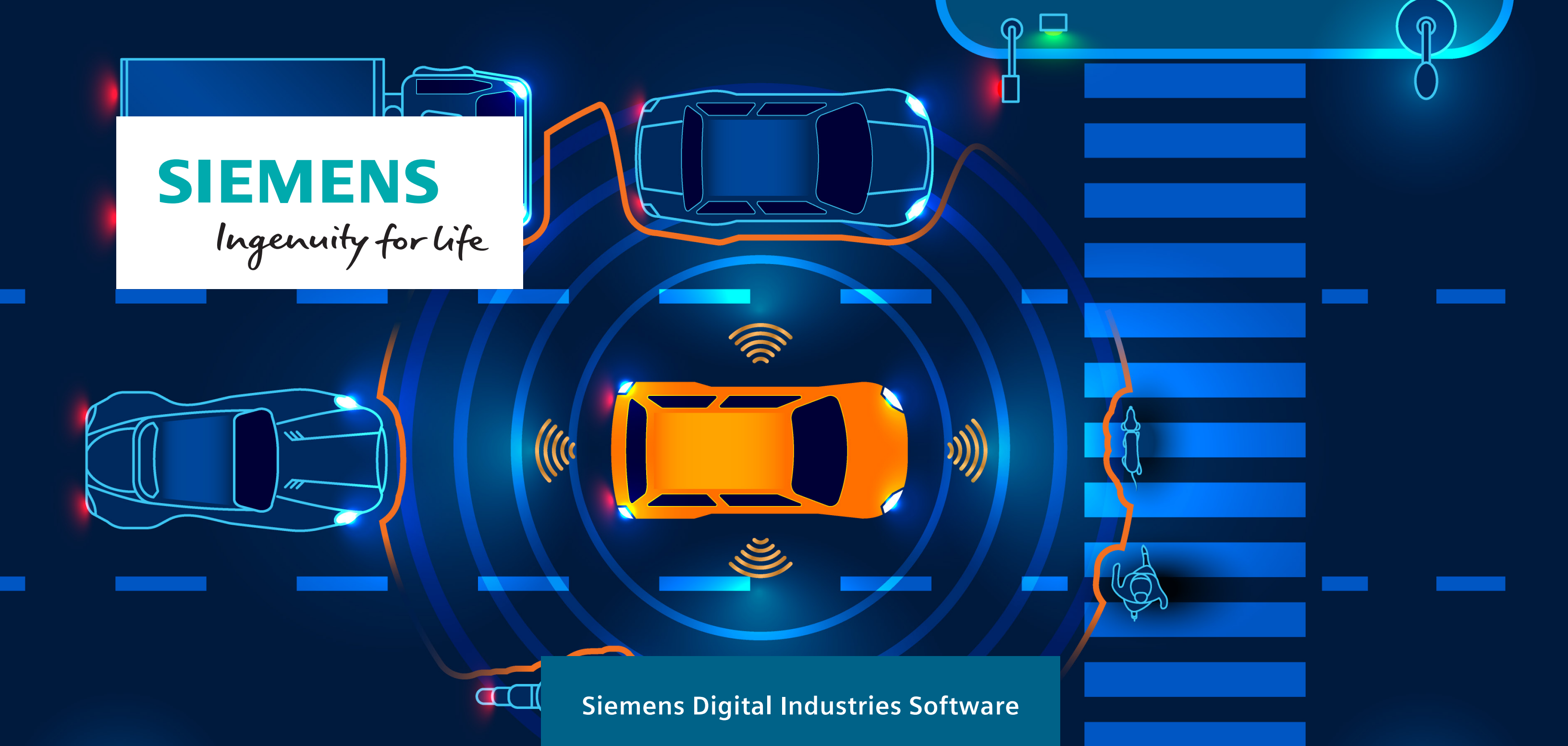

Siemens Digital Industries Software

Model-based and data-driven learning control for safety and comfort for autonomous driving

Executive summary

This paper presents developments in autonomous vehicle control exploiting combination architectures of model-based and machine learning to enhance both safety and comfort performance. Recently, machine learning has been investigated in ADAS control; however, the disadvantages are lack of rigorous results on explainability and safety. We discuss several strategies that can incorporate data learning in controls development dealing with these challenges. The presented use cases include imitation learning to learn human-like driving in lane keeping; Gaussian process adaptive control to predict vehicle states in snowy driving; reinforcement and iterative learning control; and safety verification of neural networks. We also present several testing methods with model- and hardware-in-the-loop for testing and validation.

Tong Duy Son, Ajinkya Bhave, Wouter Vandermeulen, Theo Geluk, Matthieu Worm, Herman Van der Auweraer

Contents

Introduction 3

Control technologies development......................... 4
Nonlinear MPC with long prediction horizon to increase safety and comfort performance 4
Learning adaptive control to improve safety against disturbance 5
Gaussian processes for learning and control 5
L1-adaptive control.. 6
Learning from experience to improve control performance of similar driving tasks 8
Reinforcement learning with a formal safety envelope 8
Imitation learning for human-like driving.................... 9

Testing and validation 10
Model and hardware in the loop (MiL/HiL) 10
Virtual sensing 11
Verification of neural networks 12

Conclusions and further work 13
Innovation from chip to city 13

Conclusion 14
Acknowledgments 14

References 14

Introduction

Safety is considered the most important factor and motivation of autonomous driving (AD) and advanced driver assistance system (ADAS) development. While several assistive functionalities in the automotive industry can assist drivers in normal scenarios (such as adaptive cruise control or autonomous parking) it is desirable that future ADAS/AD systems can also deal with more safety-critical scenarios. For example, the Euro NCAP 2018 Automated Driving Tests report[1] indicates that cut-in scenarios are considered one of the most challenging tests for highway assist systems. In the cut-in test, a car from the adjacent lane merges into the lane just in front of the test car, which is a common scenario in everyday traffic. Several test vehicle ADAS controllers have failed to react sufficiently early to avoid accidents during the tests. Other examples include avoiding collision at high speed within short distance and driving in low friction conditions due to rain or snow. These types of scenarios imply challenges for model-based control such as nonlinearities, model uncertainties, environment disturbance and limited computation time. It has been shown that even an advanced model-based control design may fail to achieve safety in these scenarios of avoiding collision.[2, 3]

Comfort, or the occupant's perception of the vehicle performance in ADAS scenarios, is another challenge for automotive OEMs. Customers will only accept the ADAS functions if they experience comfortable feelings, and do not urge to take over vehicle control. Though significant knowledge is available on the performance perception for human drivers, these previous studies are no longer applicable for ADAS scenarios where the focus is on the occupant. Not being in control (steering, throttle, brake) of the vehicle as occupant, will result in different subjective assessments of either the lateral or vertical dynamics of the vehicle, and hence requires different objective performances to be tracked related to those of occupant assessments. Using those objective performances in the requirement definition for ADAS functionality enables differentiation of metrics in performance, while maintaining safety as a hard constraint to be satisfied. Ideally, in normal scenarios, the ADAS algorithms should be designed to optimize both safety and comfort performance. In critical driving scenarios, safety performance is the highest priority and should be incorporated as a hard constraint, while comfort performance could be treated as soft constraint, that is, not compulsory.

The autonomous vehicle control design usually relies on model-based control methods types, for example pure pursuit, H-infinity, linear-quadratic regulator (LQR), and model predictive control (MPC).[4] However, conventional model-based controllers have shown difficulties in dealing with safety in critical driving scenarios and in incorporating comfort objectives. Recently, several machine learning-based methods have been exploited for autonomous driving control; however, their disadvantages are lack of rigorous results on explainability and safety. In this paper, we aim to exploit the advantages of both types, sometimes in a combined architecture, to enhance safety and comfort driving performance, while still trying to guarantee formal requirements and explainability. In the first part of the paper, we discuss several technologies to improve safety and comfort driving performance. The techniques include both model-based and machine learning-based approaches:

- Nonlinear MPC with long prediction horizon to increase safety and comfort performance
- Learning adaptive control to improve safety against disturbance
- Learning from experience to improve control performance of similar driving tasks
- Reinforcement learning with a formal safety envelope
- Imitation learning for human-like driving

In the second part of the paper, we present several testing methods used for validating the proposed control developments.

Control technologies development

In this section, the developed control algorithms will be formulated in more detail, including main ideas and some initial results. The validation results are mostly

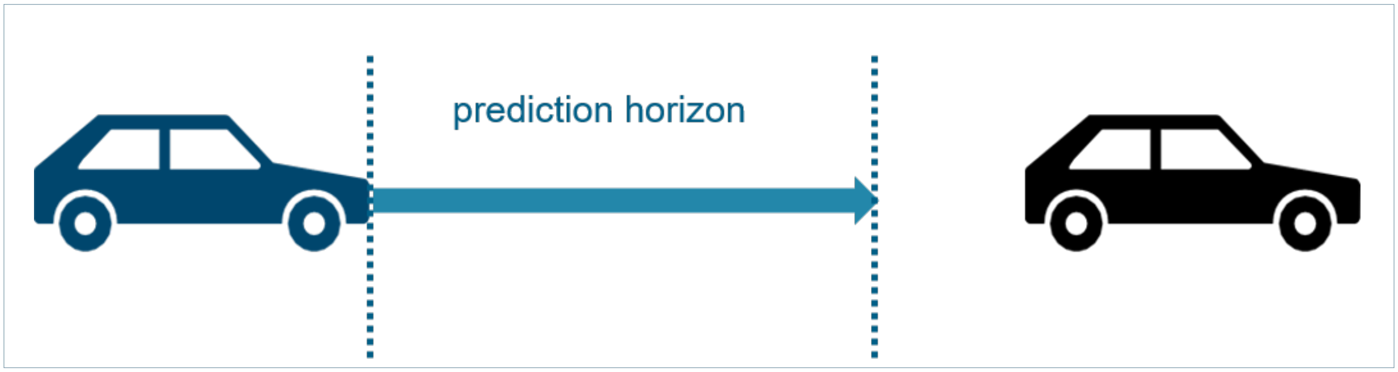

Figure 1: Prediction horizon in MPC control.

conducted in a co-simulation framework of Siemens Simcenter™ Amesim™ software and Simcenter™ Prescan™ software, which simulate high fidelity of vehicle dynamics and traffic/sensor models, respectively.

Nonlinear MPC with long prediction horizon to increase safety and comfort performance

Model predictive control (MPC) is a potential solution for autonomous driving, for both motion planning and trajectory following control. MPC design solves a model-based optimization problem over a given prediction horizon to generate control signals. MPC has been applied successfully in different industries such as chemical plants and oil refineries. And with the recent advancement in both theory and algorithms for solving optimization problems in a real-time environment, MPC controllers have been shown also capable for fast dynamic systems.[5] In autonomous driving, MPC can realize several control functionalities such as adaptive cruise and lane-keeping control while capable of incorporating vehicle dynamics, constraints (i.e. steering, throttle capabilities) and traffic environment information (i.e. safety constraints to avoid collision with other vehicles). In addition, other comfort objectives that relate to acceleration and jerk constraints can also be included. One of the main challenges of MPC is computational cost, as it relies on solving a nonlinear optimization problem each sampling.

The predictive horizon length is dependent on computation capability; consequently, the safety conditions are usually guaranteed only within a limited finite horizon. The short prediction horizon is challenging to deal with critical driving maneuvers and could lead to limited comfort performance. We have developed a nonlinear MPC algorithm to ensure safety within infinite horizon formally, based on control barrier function (CBF). Similar to Lyapunov functions to certify stability properties of a set without calculating solution of a system in nonlinear control theory, CBFs were introduced recently to certify forward invariance of a set using barrier function without computing the state's reachable set.[6] Here, we incorporate CBF into the MPC design formulation. The objectives are to guarantee safety in infinite prediction horizon and hence improve overall MPC performance. In addition, since the algorithm considers long prediction horizon, the benefits are also on comfort enhancement. That is, the car can do smooth accelerating and decelerating motion rather

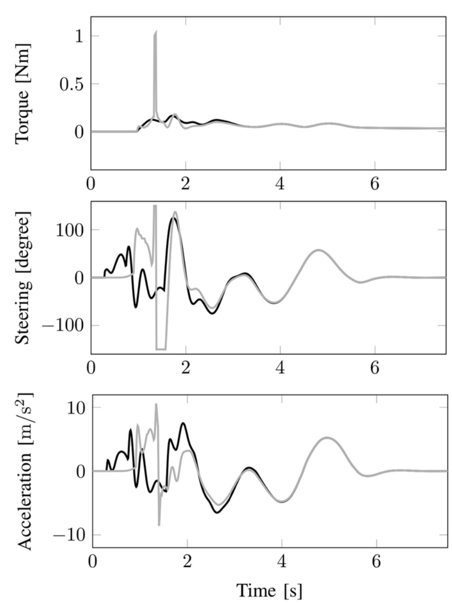

Figure: 2: Comparison between our CBF-NMPC design (black) and conventional MPC design in a critical double lane change scenario (gray): torque, steering, and lateral accelerations of car body center of gravity.

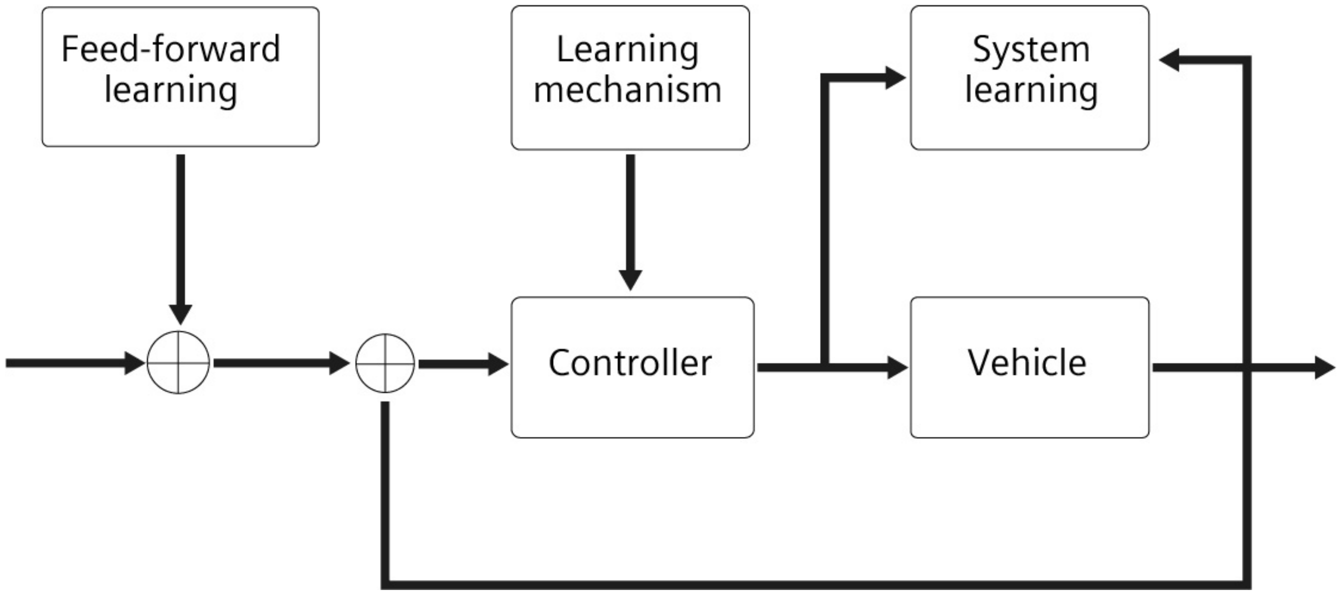

Figure 3: Adaptive control architecture.

than sudden/non-smooth ones. As an example, figure 2 presents results from a safety-critical scenario control where the car does double lane change maneuvers to avoid a static object in short distance. The figure illustrates control (torque/steering) and acceleration signals from our control design and conventional nonlinear MPC. It is clearly seen that our design obtains non-saturated and smooth control actions. More technical details can also be found in reference 3.

Learning adaptive control to improve safety against disturbance

One of the main challenges of model-based control design is that it requires a reasonably accurate but low-order mathematical model (transfer function or state space) of the system. In safety-critical scenarios, the system dynamics are however often too complex or computationally expensive to be incorporated. On the other side, the required model development and engineering efforts to deal with various complexities would pose significant difficulty and limit the use of physics-based models in model-based control.

In this work, we attempt to learn these complex dynamics from data learning and representation. The combined physics-based and data learning-based model is then exploited for a combined control architecture, shown in figure 3. Two different model learning methods are developed based on Gaussian Processes and L1-Adaptive control.

Gaussian processes for learning and control

Gaussian process (GP) is investigated for modeling dynamical systems. Gaussian process regression is a statistical machine learning method. Given a training set containing inputs and their corresponding measured output values, GP regression aims to learn a function that predicts the output at new, unobserved input locations. GPs are flexible and able to capture nonlinear dynamics with fewer parameters and less computation than other machine learning methods like deep learning. It requires less training data to obtain a reasonable prediction, in particular for control purposes.[7, 8] Moreover, the GP model provides quantification of its prediction uncertainty, which is essential to analyze robustness and safety aspects. Because of these properties, GPs are investigated for online control as a tool to handle complex dynamics. Figure 5 illustrates the advantage of GP in predicting the vehicle dynamic on low road friction disturbance, as well as providing its uncertainty confidence level. It is seen that accounting for GPs to learn the unknown dynamics is beneficial to predicting the vehicle dynamics, compared to using only low-order mathematical dynamic models.

In more detail, we formulate GPs for learning road disturbances in the form of varying road friction and incorporate GPs into a nonlinear MPC design formulation. The proposed GP-MPC technique is validated with a double lane change scenario when the tire road friction is low – as commonly seen in snowy/rainy

Figure 4: Lane change driving in snowy weather.

weather. As expected from better dynamics prediction, we have shown that GP-MPC outperforms traditional MPC in handling safety, but with more computational cost.[9]

L1-adaptive control

The autonomous truck is one of the first potential significant markets for self-driving technologies, promising increased safety, improved productivity and lower costs. Truck and logistics companies are considering the economic benefits of autonomous driving functionalities, primarily because of the numerous inefficiencies related to human drive that could be avoided (rest breaks, holidays, etc.). Autonomous trucks can work around the clock, surpassing the performance of a human-driven truck. Still, truck automation faces various challenges: in particular, autonomous trucks must demonstrate reliability and feasibility within the operations that the human-driven trucks can perform. A truck is used in varying conditions, often difficult, for thousands of kilometers. Therefore, the autonomous truck should be able to withstand wind, cold temperatures, vibrations and different road conditions (ice, salt, etc.).

Many safety-critical scenarios must be considered, such as unpredictable aerodynamics effects that can arise from truck platooning or driving in a tunnel, or even possible sudden braking or steering system failures.

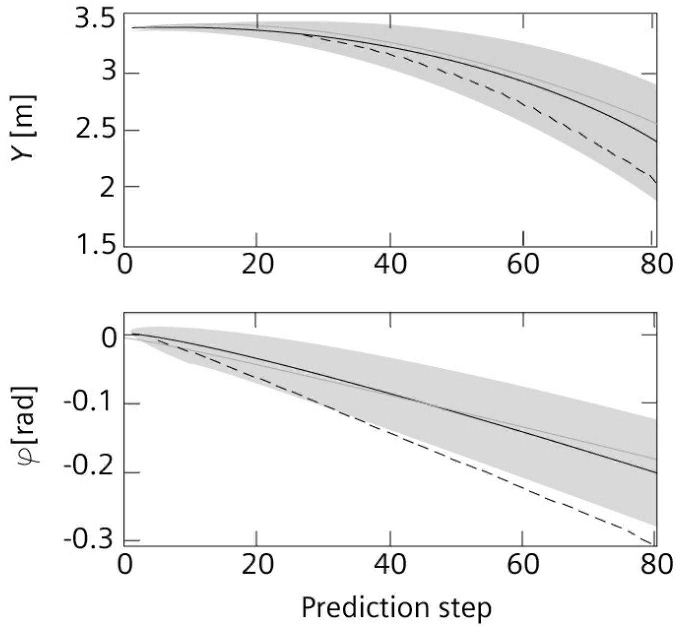

Figure 5: Predicted state evolution of y-position and vehicle heading angle using Gaussian process and nominal model compared to the real states. The uncertainty interval is in gray.

Here, we investigated L1-adaptive control to compensate for the unknown disturbance acting on the vehicle dynamic system. The key feature of L1-adaptive control compared to conventional adaptive control is the decoupling of the adaptation loop from the control loop, which enables arbitrarily fast adaptation without sacrificing robustness. The controller performance is evaluated for lateral wind disturbance and steering rack failure where only a fraction of the steering input computed by the controller reaches the vehicle. Hence, we assess the controller capability to adapt to a sudden disturbance and actuation failure. The results are shown in figure 7 for a roundabout driving scenario. They demonstrate the higher safety and adaptable capability performance of the proposed L1-adaptive mechanism. With different levels of disturbances, the vehicle tracking performance remains stable.

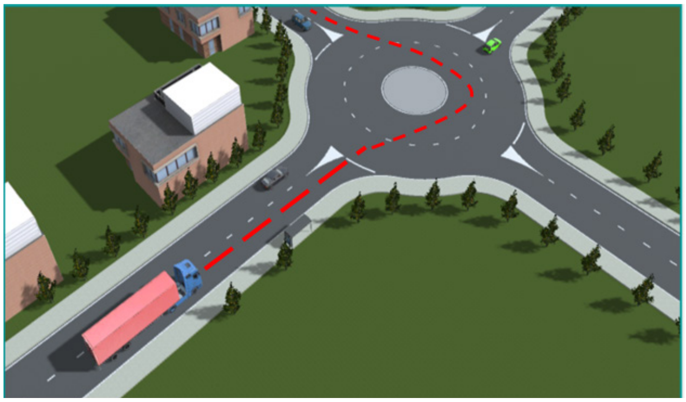

Figure 6 : Autonomous truck on a roundabout.

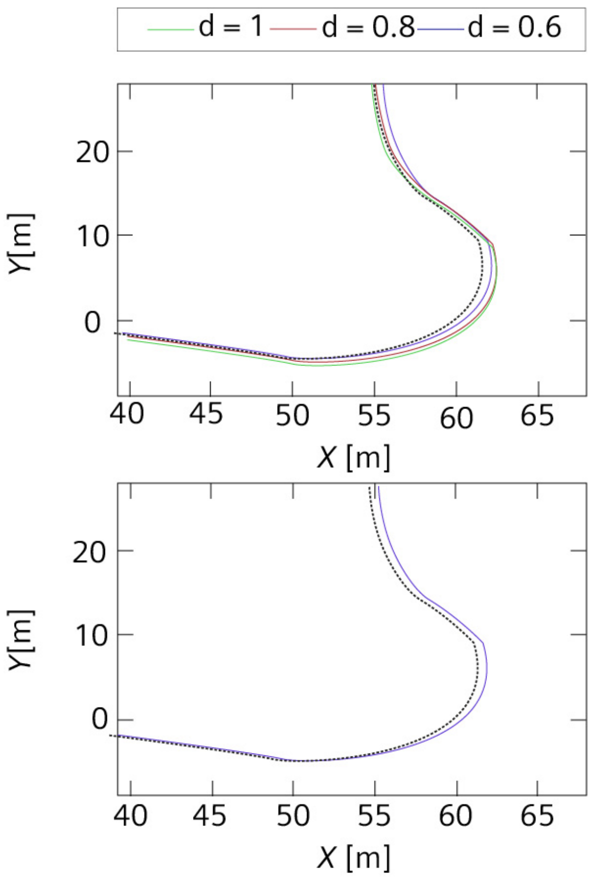

Figure 7: Roundabout driving performance with respect to different disturbance parameters – upper: conventional control; lower: L1-adaptive control. The dashed line is the reference trajectory.

Figure 8: Drift parking scenario: the blue car tries to park between two cars.

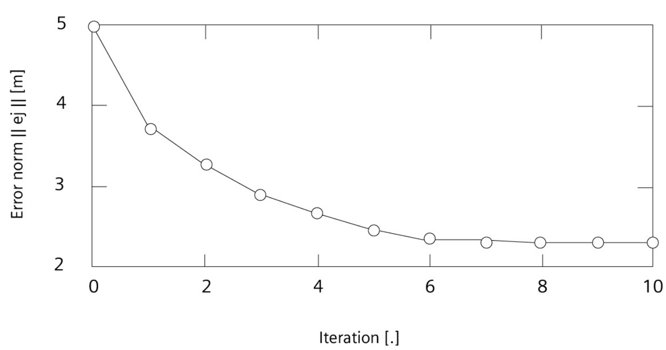

Figure 9: Tracking error performance in the iteration domain.

Learning from experience to improve control performance of similar driving tasks

Next, we apply a specific type of data learning for feedforward control, namely iterative learning control (ILC), to improve driving control performance of similar driving tasks. Instead of taking action on feedback control, the key idea of ILC is to update a feedforward control signal iteratively based on measured data from previous iterations. The essential property in ILC is system repeatability, that is, a system is required to follow an identical or similar trajectory. Although this may seem to be restrictive, many practical systems are highly repetitive, for example, racing cars, school and shuttle buses, valet parking to a predetermined spot, and testing of a specified driving scenario in standard tests (i.e. NCAP tests). Through learning in the iteration domain, ILC can achieve high tracking performance despite large model uncertainty and repeating disturbances. Rigorous stability and performance analyses are also available in ILC literature to support and explain the driving performance while learning.

The proposed ILC design in each iteration includes two stages: model-correction and control input designs. Both designs rely on optimization. In each iteration, the input signal and the error signal between the reference trajectory and the system output are stored in memory and exploited for computing the next iteration model correction and input signals. The ILC designs are applied and validated in a drift parking scenario. The results show that with just few iterations of data learning, the control performance is significantly improved despite of unknown disturbance and uncertainties in the steering system, and tire road friction. In addition, we have also applied this technique for racing application.[10]

Reinforcement learning with a formal safety envelope

Reinforcement learning (RL) is a branch of machine learning that studies the problem of training an agent in a real or simulated environment by rewarding or punishing the agent for actions taken during its interaction with the environment. This reward function formalizes the utility of each action and forces the agent to recognize an optimal action policy based on the current state of the environment that maximizes the expected future discounted rewards obtained during a task. RL is conceptually different from supervised or unsupervised learning, where the training is typically done offline on a static dataset of labeled or unlabeled examples. In RL, the training is done in an online fashion with continuous interaction between the agent and the environment. This means the dataset can constantly change during the training phase. For safety-critical scenarios, this requires a high-fidelity model or simulation of the environment, since the car (agent) cannot execute dangerous maneuvers in the real world to learn from its mistakes.

Deep RL (DRL) is the extension of classical RL techniques with deep neural networks (DNNs) to learn specific functions in the framework. DNNs have been used for learning different value functions: the state (V) or the state-action (Q) function, in model-free RL, as well as for capturing the detailed nonlinear dynamics of the plant in model-based RL. DRL is a constantly evolving, diverse domain with multiple algorithms and paradigms effectively being used to solve complex problems in robotics and autonomous driving. A compact survey on RL state-of-the-art is available.[11]

We use DRL for multiple applications in autonomous driving, from learning different driving styles from vehicle controller area network (CAN) and raw sensor data to developing RL-based controllers with a formal safety envelope for common ADAS maneuvers (lane keeping and vehicle following). As an example, in figure 10 we show an RL-trained lane-following controller that also keeps a safe distance from the lead vehicle, based on a formal safety standard, Responsibility-Sensitive Safety (RSS) from Intel.[12]

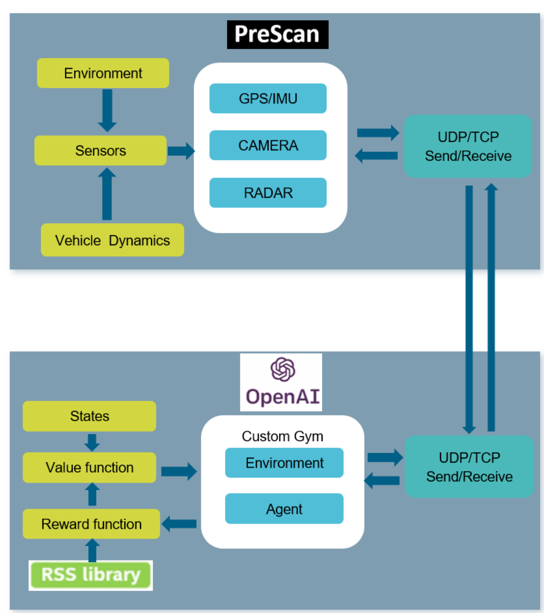

Figure 10: OpenAI reinforcement learning framework coupling with Simcenter Prescan.

Our development framework consists of the Simcenter Prescan simulator connected to the popular OpenAI Gym environment for RL development,[13] with the RSS library integrated into the training loop. Coupling a high-fidelity simulator like Simcenter Prescan to an open RL training framework like Gym allows engineers to leverage the latest algorithm advances in RL with accurate vehicle, sensor and environment models to enable rapid development of deep learning-based controllers. During the training loop, Simcenter Prescan sends to the Gym module the ego and lead vehicle dynamic state and actuator signals, along with images from a front-facing camera at full resolution. The images are processed by a deep neural network (DNN) to estimate the ego vehicle lateral position based on the lane markings. This estimate is combined with ego vehicle state and the distance to the front vehicle to train a DRL-based controller in Gym to keep lanes and follow the front car, while keeping a verifiably safe distance calculated from the RSS standard. For this canonical example, we use a deep deterministic policy gradient (DDPG)[14] approach for continuous and precise control of the ego vehicle to successfully train a driving policy in closed-loop simulation.

Imitation learning for human-like driving

Finally, we study imitation learning technique as a guidance for control. Imitation learning or end-to-end learning has been considered significantly recently as a supervised learning approach to learn from driving data.[15] The main idea is to create a policy that mimics the driving behaviors of (good) human drivers. The common challenges of imitation learning are the large datasets required, and mismatch between the distribution of training and test data. In addition, the explainable safety and control properties (i.e. stability, settling time, overshoot) of the neural network are still largely understudied.

In this part, we exploit the advantages of both model-based and machine learning-based approaches for a hierarchical mid-to-mid framework. The inputs to the proposed learning model are representative features coming from processed sensor data and the outputs are the reference trajectories. As a result, the data-driven learning outcome can be treated as a motion planning layer. The imitation learning training model is implemented using online dataset aggregation (DAgger).[16]

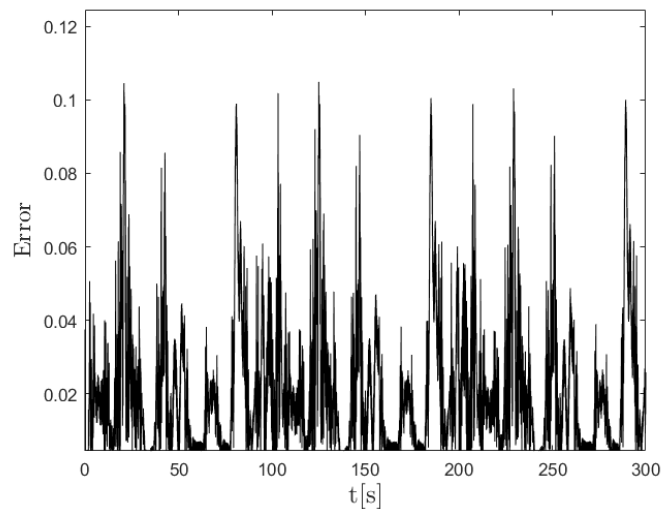

Figure 11: Euclidean distance error between the ACPN and the expert on a 300s trajectory after the 10th rollout.

This is an iterative supervised learning fashion, with an increasing dataset due to the exposure of the expert driver to new states induced by the learner. The dataset for learning is generated from virtual Simcenter Amesim and Simcenter Prescan data (see figure 11). In the training loss function of the neural network, we incorporate from the beginning the knowledge of safety objectives such as collision avoidance with road boundaries through barrier function constraints. Combined with DAgger, the proposed loss function will show advantages on both safety improvement and convergence speed. In addition, we propose to use B-spline trajectory parametrization, choosing spline coefficients instead of points as output nodes for learning. The advantage of this method is that a B-spline is always contained in the convex hull of its coefficients. Therefore, safety constraints on generated trajectories can be imposed by only constraining the spline coefficients in the barrier function of the loss function. This technique improves computation and learning efficiency aspects.

The proposed development is validated in a lane-keeping scenario. Figure 12 and figure 13 demonstrate the Euclidean error to the lane center and the vehicle trajectory, respectively, after the 10th rollout. The results show that the vehicle can follow the lane center using the proposed algorithm. More details are given in our recent work.[17]

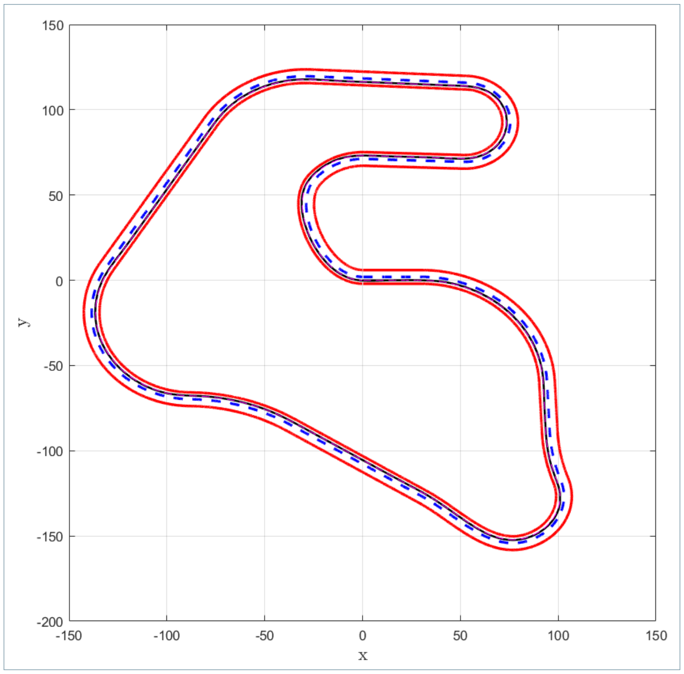

Figure 12

Testing and validation

In this section, we discuss several testing technologies exploited in our works for validating the proposed algorithm developments toward safety and comfort objectives.

Model and hardware in the loop (MiL/HiL)

We first discuss briefly the motivation of using simulation for model-in-the-loop and hardware-in-the-loop testing and their roles in safety and comfort development.

It is recognized in the autonomous driving industry that simulation is an efficient method for testing and validating ADAS/AD functionalities. The traffic environment has a wide variety of parameters from road types, vehicles, pedestrians, cyclists, obstacles and weather. The number of scenarios grows exponentially with the number of parameters and can easily explode up to millions Furthermore, all scenarios cannot be produced and reproduced easily in real life, and real road testing is valid for a specific mechanical, electrical and software configuration. If needed to adapt to a new configuration or software update, the test must be conducted again. Therefore, the major part of the ADAS functionalities will be validated through simulations.

Furthermore, MiL/HiL testing with respect to safety is particularly essential for several additional reasons:

- Physical vehicle testing for safety-critical scenarios is dangerous and expensive. An incomplete algorithm being deployed in a physical car can always cause collision.
- Development of data-driven based algorithms like machine learning and imitation learning requires datasets for training the algorithm. It is however very difficult and expensive to have datasets of scenarios relevant to safety, for example, accident or near-accident, collision avoidance and lane change at high speed.

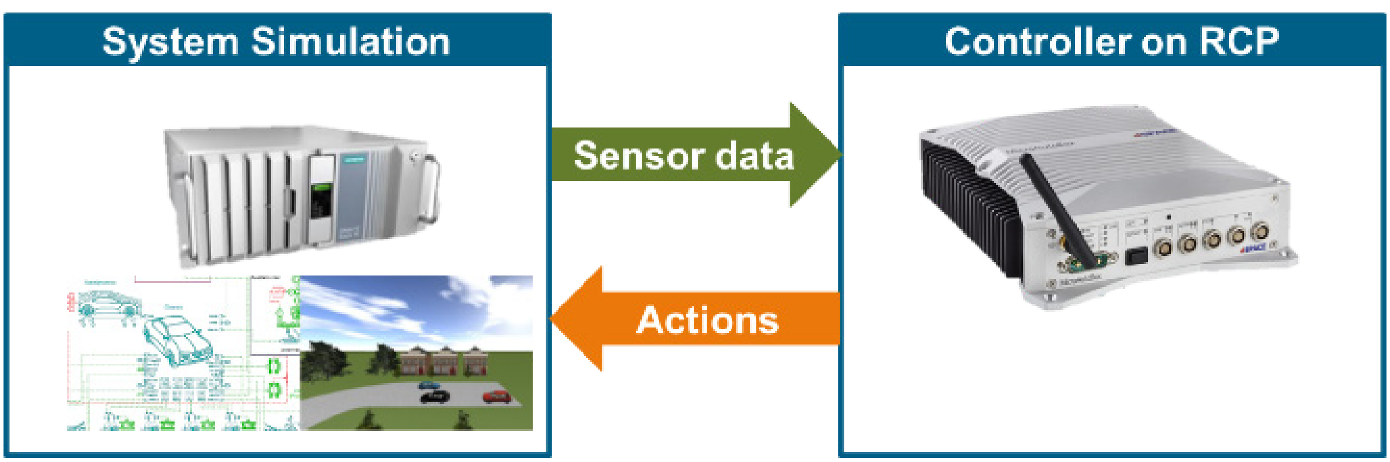

Figure 13

On the comfort side, while it is still not clear to use MiL/HiL for subjective comfort, these tests are certainly relevant to objective comfort, for example, in analyzing accelerations, jerk and time-to-collision features.

Figure 14: Different autonomous vehicle platforms: small vehicles with augmented reallity (AR), Simrod vehicle, and conventional vehicle equipped with driving robots.

Several steps need to be taken before an ADAS controller can be tested on the vehicle in a real-life environment. A first step after the offline evaluation and simulation is to evaluate the performance of the controller on a rapid control prototyping (RCP) platform. The vehicle model is a Simcenter Amesim vehicle dynamics model, the environment and sensors are simulated in Simcenter Prescan. These two platforms are linked to simulate the performance of the controller on a vehicle (see figure 13).

A next step is to deploy the developed controller onto an embedded system and evaluate if the controller is still performing as designed. In this case the dSPACE rapid control prototyping system is replaced with the controller on an embedded system. The rest of the setup is kept as was for the previous step: the vehicle, environment and sensors can be simulated on a real-time platform (see figure 14). This approach enables a robust evaluation of the controller by a simulation-based vehicle and environment. Measurement noise and system uncertainty can be part of defined test conditions.

Finally, figure 14 illustrates some of our testing setups for deployments on a physical vehicle with autonomous driving sensors (Lidar, camera, radar) and localization system (GPS and inertial measurement unit - IMU). The setups range from lab-based testing environment to proving ground track testing environment levels, providing great flexibilities during solution development.

Virtual sensing

Virtual sensing plays an important role in the development of controls, particularly for learning adaptive control types presented above in the "Learning adaptive control to improve safety against disturbance" section. Physical quantities that are difficult, impossible or expensive to measure can be estimated with approaches such as Kalman filters that use system models to estimate system states. In most vehicles there are sensors available that measure or estimate the vehicle motion (center of gravity position and velocity). However, this centralized measurement point does not allow an accurate evaluation of how an occupant would perceive the motion of the vehicle.

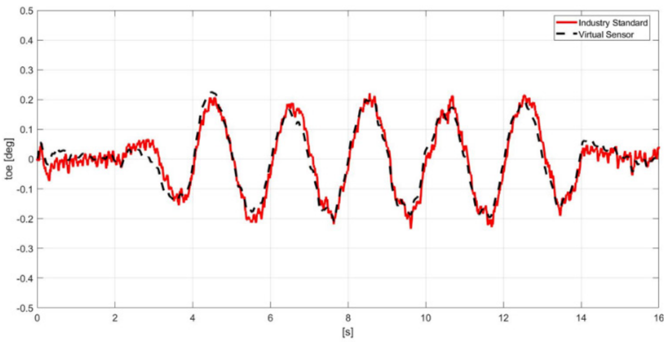

Figure 15: Virtual sensing result.

The subjective evaluation of the occupant of the vehicle is much more determined by local responses.[18]

Siemens uses virtual sensing with cost-efficient and easy-to-install sensors to estimate the localized responses of the suspension. These responses are crucial input for the controller to make sure the comfort of the vehicle is maximized while still maintaining the safety requirements.

Figure 15 shows the accuracy obtained when using a virtual sensor to estimate the localized responses. The virtual sensor result (dashed black line) is compared with a standard and expensive sensor output (red line) that measures the motion of the front wheel. As can be seen the virtual sensor can estimate this local response accurately, while keeping the total solution cost price lower than the standard sensor. The subjective evaluation of the vehicle motion is determined heavily by very small differences; accurate measurements of the localized responses are crucial to enable a correct control action to maximize comfort.

Verification of neural networks

The biggest impediment to the acceptance and integration of DNNs in production automotive systems is the inability to formally verify the decisions reached by DNNs. During training of DNNs, a separate dataset of (possibly) labeled examples is used to teach the network the relationship between inputs and expected outputs. A separate validation dataset is used during training to tune the model's hyper-parameters and to prevent overfitting of model parameters. Once the training is complete, a test dataset is used to quantify the generalization properties of the trained DNN on unseen data, often from a different set of scenarios that are not present in the training dataset. If the difference between the training and test accuracy is small, the network is assumed to have trained reasonably well and should generalize to unseen data.

This method of verifying the performance of a DNN makes a critical assumption: the statistical properties of the test data and of the real-world data that the DNN will encounter during operation are similar. If this assumption holds, then we can trust the DNN to have performance close to what was observed during inference in the lab. However, the real world is constantly changing because of stochastic factors and unmodeled dynamics, which makes the previous assumption unverifiable in practice. In addition, DNNs are particularly susceptible to adversarial attacks,[19] where an attacker can perturb the input to the network in a way that is imperceptible to humans but that can cause the DNN to output catastrophically erroneous predictions.

DNNs are black-box, high-capacity function approximators that are not amenable to mathematical guarantees about their performance, in general. However, there have been several advances in formally verifying properties of DNNs under certain assumptions about their architecture and activation functions. A class of DNNs that has been researched extensively are the feed-forward networks with rectified linear units (ReLUs) as activation functions. Verifiable properties include specifications that hold over the entire input space, guarantees over the states reached in the output space using reachability analysis, bounded input-output relations for DNNs to guarantee closed-loop stability of feedback loops, and invariant properties over time for recurrent neural networks (RNNs). The area of formal verification of DNNs is a rapidly evolving field with the promise for scalable application to industrial networks in the future. A survey of recent approaches and tools available for such analysis of DNNs is available.[20]

Explainable AI (XAI) is an area of research that studies how to enable interpretation of predictions by a DNN in a way that is easily understood by human collaborators. The focus is on machine learning techniques that allow human users to understand, appropriately trust and efficiently deploy state-of-art DNNs in safety-critical or production environments.[21] Many of the approaches used in XAI are also applicable to verification of DNNs, since the common aim is to understand and predict the performance of such models with some notion of formal guarantees. The XAI approach is also relevant for functional safety standards like ISO 26262 and SOTIF, which currently do not have a well-mapped strategy to verify the specifications and operating design domain (ODD) for components based on machine learning algorithms, such as perception and motion planning modules in autonomous vehicles.

Conclusions and further work

Safety and comfort designs and balancing are essential objectives in realizing autonomous driving. Several control methods have been proposed to tackle these challenges. On one side, we try to advance conventional control and data-driven techniques such as MPC or imitation learning, and on the other side, these techniques can also be combined to exploit advantages of both worlds. Moreover, several testing methods have also been discussed. An efficient testing process that can reduce cost and effort is very valuable, and also significantly reduces the amount of real prototype tests that must be performed.

Innovation from chip to city

As a global industry leader, Siemens has a clear focus on innovation. With regards to transportation and mobility, Siemens delivers pioneering technologies that will radically change mobility in the near future, enabling electrification, autonomous driving, smart cities and more. With the Siemens Digital Industries Software solutions portfolio, manufacturers can deploy a digital twin approach from chip to city to bring complex, smart products to market faster and with greater confidence. Starting at the level of the chip, the Siemens Mentor portfolio simulates and emulates chip designs and delivers a validation and verification environment that supports realistic and application-specific testing to ensure security, functional safety, low power consumption and maximum performance of core electronics. At the level of the embedded systems architecture, Siemens Capital software enables integral electric, network and software architecture design and implementation. Siemens Simcenter solutions deliver simulation environments to optimize safety and comfort performance at system and full vehicle level and make it possible to run hundred thousands of traffic scenarios virtually, but also on the proving ground. Finally, combining forces of Siemens Digital Industries Software and Siemens Mobility, mobility as a service and vehicle to infrastructure technologies are developed and installed in cities worldwide. There is a strong focus on driving continuity and traceability of requirements, architecture definitions, simulation models and test results throughout the entire supply chain, truly stretching from chip to city.

To optimize the safety and comfort performance of autonomous vehicles, Siemens promotes a closed-loop vehicle development process that consumes recorded data during the lifecycle of the vehicle to drive improvements in the design of the vehicle and its controllers. A framework for continuous virtual and physical validation and verification enables frequent over-the-air software updates and regular hardware improvements. Simcenter offers the services, tools and methods to capture and crunch engineering and scenario data. Consequently, it offers intelligence in simulation, supporting generative design and turnkey massive validation and verification.

Conclusion

Acknowledgments

This work has been conducted within the Flanders Make ICON Integrated Identification for Control (ID2CON) and ITEA3 Embedded systems with physical models in the production code software (EMPHYSIS) projects. For both projects, financial support from the Flemish Agency for Innovation and Entrepreneurship (VLAIO) is gratefully acknowledged.

References

1. https://www.euroncap.com/en/vehicle-safety/safety-campaigns/2018-automated-driving-tests/
2. J. Nilsson, Y. Gao, A. Carvalho, and F. Borrelli, "Manoeuvre generation and control for automated highway driving," in 19th IFAC World Congress, vol. 47, pp. 6301–6306, 2014
3. T. D. Son and Q. Nguyen, "Safety-critical control for non-affine nonlinear systems with application on autonomous vehicle," in 58th IEEE Conference on Decision and Control, Nice, 2019.
4. Automated Driving: Safer and More Efficient Future Driving, Springer International Publishing, 2017
5. J. B. Rawlings, D. Q. Mayne, and M. Diehl, Model Predictive Control: Theory, Computation, and Design, Nob Hill Publishing, 2nd ed., 2017.
6. P. Wieland and F. Allgower, "Constructive safety using control barrier functions," IFAC Proceedings, vol. 40, pp. 462–467, 2007.
7. C. E. Rasmussen and H. Nickisch, "Gaussian processes for machine learning (GPML) toolbox," Journal of Machine Learning Research, vol. 11, pp. 3011–3015, Dec. 2010
8. T. X. Nghiem, "Linearized gaussian processes for fast data-driven model predictive control," 2019 American Control Conference (ACC), pp. 1629–1634, July 2019
9. R. R. Arany, T. D. Son, and H. V. der Auweraer, "Learning control for autonomous driving on slippery snowy road conditions," in IEEE Conference on Control Technology and Applications, 2020.
10. A. Steinhauser, T. D. Son, E. Hostens, and J. Swevers, "Rofalt: an optimization-based learning control tool for nonlinear systems," 2018 IEEE 15th International Workshop on Advanced Motion Control (AMC), pp. 198–203, March 2018.
11. A. B. Arrieta, N. Dıaz-Rodrıguez, J. D. Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, "A brief survey of deep reinforcement learning," Special Issue on Deep Learning for Image Understanding, IEEE Signal Processing Magazine, 2017
12. Shalev-Shwartz, S. Shammah, and A. Shashua, "On a formal model of safe and scalable self-driving cars," arXiv:1708.06374, 2017.
13. https://gym.openai.com/.
14. T. P. L. et al., "Continuous control with deep reinforcement learning," International Conference on Learning Representations (ICLR), 2016.
15. M. Bansal, A. Krizhevsky, and A. Ogale, "Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst," arXiv:1812.03079, 2018
16. S. Ross, G. Gordon, and D. Bagnell, "A reduction of imitation learning and structured prediction to no-regret online learning," Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, pp. 627–635, 2011.
17. F. S. Acerbo, T. D. Son, and H. V. der Auweraer, "Safe and computational efficient imitation learning for autonomous vehicle driving," IEEE American Control Conference, 2020.
18. C. Gagliano and T. Geluk, "Importance of body rigidity in the transient stage of the maneuver," 6th International Munich Chassis Symposium 2015, (Wiesbaden), pp. 347–357.
19. W. Wei, L. Liu, S. Truex, L. Yu, and M. E. Gursoy, "Adversarial examples in deep learning: Characterization and divergence," Computing Research Respositoy (CoRR), vol. abs/1807.00051, 2018.
20. C. Liu, T. Arnon, C. Lazarus, C. W. Barrett, and M. J. Kochenderfer, "Algorithms for verifying deep neural networks," Computing Research Respositoy (CoRR), vol. abs/1903.06758, 2019.
21. A. B. Arrieta, N. Diaz-Rodriguez, J. D. Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, "Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible AI," Information Fusion, vol. 58, pp. 82 – 115, 2020

Siemens Digital Industries Software

Headquarters
Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 972 987 3000

Americas
Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 314 264 8499

Europe
Stephenson House
Sir William Siemens Square
Frimley, Camberley
Surrey, GU16 8QD
+44 (0) 1276 413200

Asia-Pacific
Unit 901-902, 9/F
Tower B, Manulife Financial Centre
223-231 Wai Yip Street, Kwun Tong
Kowloon, Hong Kong
+852 2230 3333

About Siemens Digital Industries Software

Siemens Digital Industries Software is driving transformation to enable a digital enterprise where engineering, manufacturing and electronics design meet tomorrow. Xcelerator, the comprehensive and integrated portfolio of software and services from Siemens Digital Industries Software, helps companies of all sizes create and leverage a comprehensive digital twin that provides organizations with new insights, opportunities and levels of automation to drive innovation. For more information on Siemens Digital Industries Software products and services, visit siemens.com/software or follow us on LinkedIn, Twitter, Facebook and Instagram. Siemens Digital Industries Software – Where today meets tomorrow.

siemens.com/software

82192-C4 8/20 K