Towards Mechatronics Approach of System Design, Verification and Validation for Autonomous Vehicles

Chinmay Samak*, Tanmay Samak*, Venkat Krovi

Abstract—Modern-day autonomous vehicles are increasingly becoming complex multidisciplinary systems composed of mechanical, electrical, electronic, computing and information sub-systems. Furthermore, the individual constituent technologies employed for developing autonomous vehicles have started maturing up to a point, where it seems beneficial to start looking at the synergistic integration of these components into sub-systems, systems, and potentially, system-of-systems. Hence, this work applies the principles of mechatronics approach of system design, verification and validation for the development of autonomous vehicles. Particularly, we discuss leveraging multidisciplinary co-design practices along with virtual, hybrid and physical proto-typing and testing within a concurrent engineering framework to develop and validate a scaled autonomous vehicle using the AutoDRIVE Ecosystem. We also describe a case-study of autonomous parking application using a modular probabilistic framework to illustrate the benefits of the proposed approach.

Index Terms—Autonomous vehicles, mechatronics approach, multidisciplinary design, simulation and virtual prototyping, rapid prototyping, verification and validation.
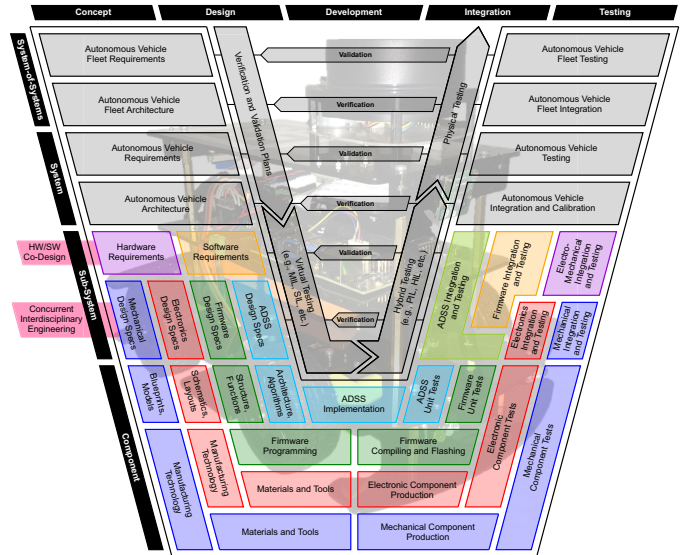
Fig. 1. Extended V-model fostering mechatronics approach of system design, verification and validation for autonomous vehicles. The model depicts evolution of a concept into a product through decomposition, design, development, integration and testing across component, sub-system, system and system-of-systems levels in a unified concurrent interdisciplinary engineering framework.

I. INTRODUCTION

AUTOMOTIVE vehicles have evolved significantly over the course of time [1]. The gradual transition from purely mechanical automobiles to those with greater incorporation of electrical, electronic and computer-controlled sub-systems occurred in phases over the course of the past century; with each phase improving performance, convenience and reliability of these systems. Modern vehicles are increasingly adopting electrical, electronic, computing and information sub-systems along with software algorithms for low-level control as well as high-level advanced driver assistance system (ADAS) or autonomous driving (AD) features [2]. This naturally brings in the interplay between different levels of mechanical, electrical, electronic, networking and software sub-systems among a single vehicle system, thereby transforming them from purely mechanical systems, which they were in the past, to complex multidisciplinary systems [3]. As such, while it may have been justifiable for earlier ADAS/AD feature developers to focus on core software development, the increasing complexity and interdisciplinary nature of modern automotive systems can benefit from synergistic hardware-software co-design complemented with integrated verification and validation by following the mechatronics principles.

*These authors contributed equally.

C. V. Samak, T. V. Samak and V. N. Krovi are with the Automation, Robotics and Mechatronics Laboratory (ARMLab), Department of Automotive Engineering, Clemson University International Center for Automotive Research (CU-ICAR), Greenville, SC 29607, USA. Email: {csamak, tsamak, vkrovi}@clemson.edu

Mechatronics engineering [4]–[6] focuses on concurrent and synergistic integration of mechanical, electrical and electronics engineering, computer science and information technology for development and validation of complex interdisciplinary systems. This ideology is derived from the fact that various components of a "mechatronic" system, often belonging to a multitude of disciplines, influence each other and hence have a design impact at the component, sub-system, system and system-of-systems levels. The resulting "mechatronic" realization now builds on capabilities endowed by the various constituent layers. In such a milieu, the system development approach has also evolved from relatively ad-hoc to the more formal V-model [7], building on the modular software development and validation roadmap [8]. This model has evolved through several progressions [9] and our work seeks to further formalize the adoption of mechatronics approach of system conceptualization, design, development, integration and testing for autonomous vehicles (refer Fig. 1).

A recent book [10] highlights best practices for industrial design, development and validation of autonomous vehicles and notes the significant adoption of model-based design (MBD) for system integration and testing. However, similar adoption of such streamlined workflows by academia has
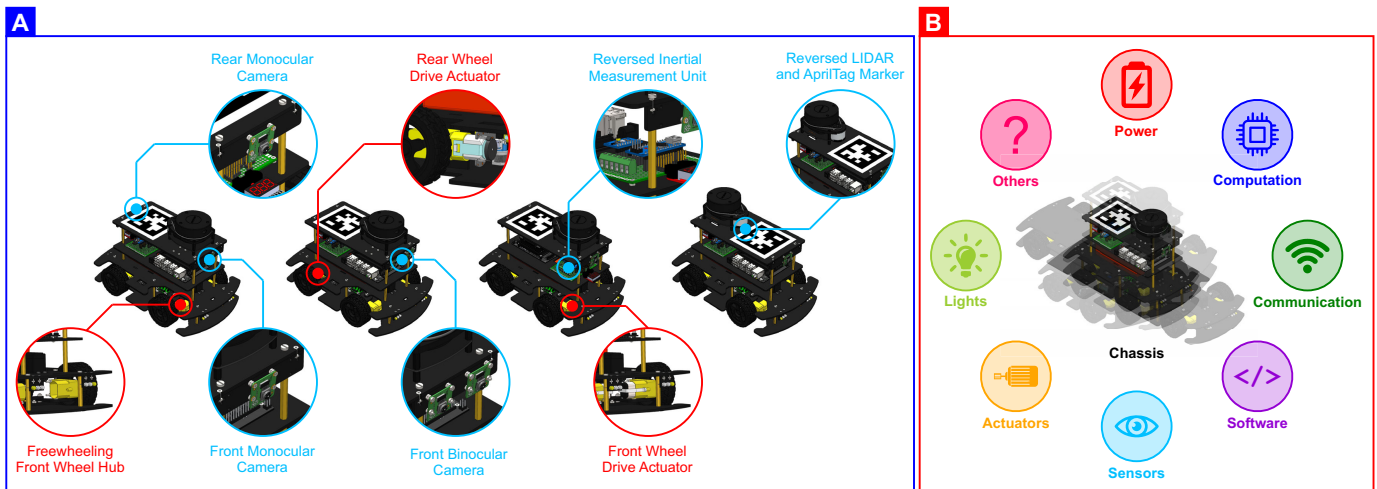
Fig. 2. AutoDRIVE Ecosystem fosters mechatronics design principles at two levels: [A] primitive reconfigurability allows permutations and combinations of addition, removal or replacement of selective components and sub-assemblies of the vehicle to better suit the application; [B] advanced reconfigurability allows complete modification of existing hardware and software architectures, and provides an opportunity for introducing new features and functionalities to the ecosystem.

lagged behind [10]. This gap could be explained by the virtue of standardization (e.g. ISO 26262 [11], ISO/IEC 33061 [12], VDI 2221 [13], VDI 2206 [14], AUTOSAR [15], etc.) in industries versus the fact that majority of academic projects are deployed using fragmented hardware-software ecosystems (e.g. hobby platforms) with a key focus on developing low-cost initial proof-of-concept implementations. Additionally, such an opportunistic and potentially uninformed selection of hardware [16]–[18] and software [19]–[21] toolchains hinders adoption of co-design and concurrent engineering thinking to full extent.

In this paper, we discuss the design philosophy and one of the key motivation factors behind AutoDRIVE Ecosystem[1] [22], [23] – a tightly integrated ecosystem for developing, simulating and deploying autonomous vehicles across scales and complexities. The said philosophy involves adopting and promoting mechatronics approach of system design, verification and validation for autonomous vehicles, with an emphasis on creating a streamlined pathway for transitioning towards the ultimate industrial practice. This paper also describes a detailed case-study, which demonstrates the methodical adoption of mechatronics approach for designing, developing and validating a scaled vehicle in the context of autonomous parking[2] application using a modular probabilistic framework.

II. MULTIDISCIPLINARY DESIGN

AutoDRIVE offers an open-access, open-interface and flexible ecosystem for autonomous vehicle development by permitting access to and alteration of hardware as well as software aspects of the multidisciplinary system design. Particularly, this ecosystem and its various modules can assist in executing each step of every phase and level in the proposed V-model (refer Fig. 1), thereby making it an apt framework for demonstrating the claims and contributions of this work.

[1]Website: https://autodrive-ecosystem.github.io
[2]Video: https://youtu.be/piCyvTM2dek

AutoDRIVE Ecosystem offers the following two levels of design reconfigurability, thereby promoting hardware-software co-design (refer Fig. 2).

- **Primitive Reconfigurability:** AutoDRIVE's native vehicle, called "Nigel", is modular enough to support out-of-the-box hardware reconfigurability in terms of adding, removing or replacing selective components and sub-assemblies of the vehicle for perception (encoders, IPS, IMU, cameras, LIDAR, etc.), computation (high/low-level), communication (wired/wireless), actuation (Ackermann/skid-steered, front/rear/all-wheel drive), illumination (head/tail-lights, turn/reverse indicators) and power (battery, power electronics) in addition to flexibly updating the vehicle firmware and/or autonomous driving software stack (ADSS) to better suit target application(s).
- **Advanced Reconfigurability:** The completely open-hardware, open-software architecture of AutoDRIVE Ecosystem allows modification of vehicle chassis parameters (different form factors and component mounting profiles), powertrain configuration (variable driving performance), as well as firmware and ADSS architecture (software flexibility).

The fundamental step in system design is requirement specification, without which the design cannot be truly validated to be right or wrong, it can only be surprising [24]. Since AutoDRIVE was intended to be a generic ecosystem for rapidly prototyping autonomous driving solutions, the requirement elicitation resulted in a superset of requirements demanded by the application case study discussed in this paper. Furthermore, being an open-access ecosystem, AutoDRIVE allows for updating the designs of various components, sub-systems and systems for expanding the ecosystem. That being said, following is a summary of functional requirement specifications for Nigel as of this version of the ecosystem.
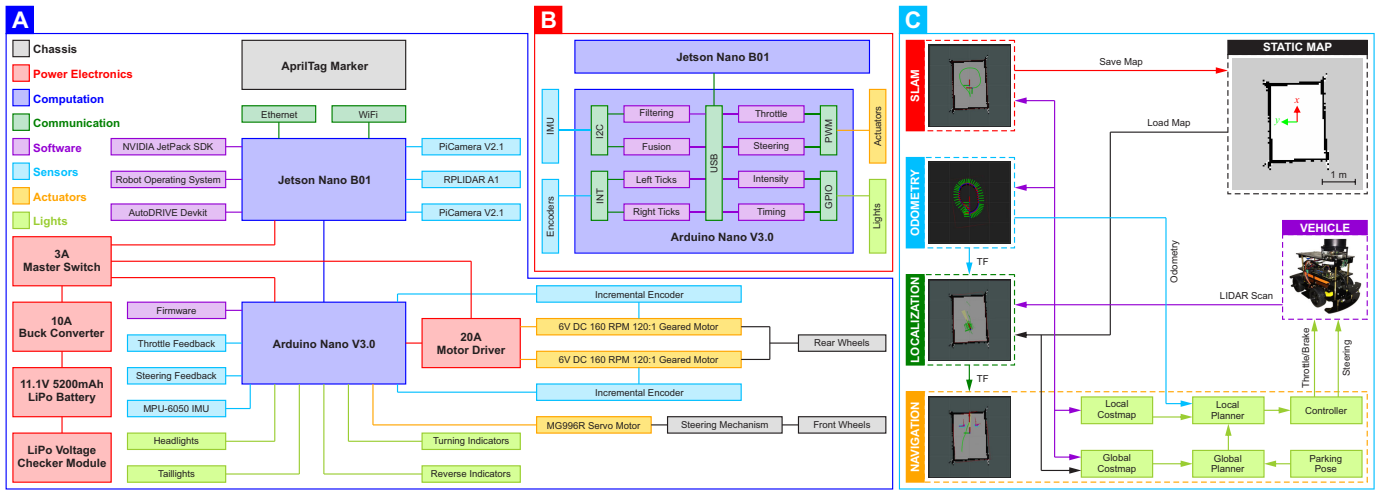
Fig. 3. Conceptualization and design of scaled autonomous vehicle: [A] hardware-software architecture; [B] firmware design specifications; [C] modular perception, planning and control architecture for autonomous parking application.

- General design guidelines:
  - Open-source[3] hardware and software
  - Inexpensive ($\leq$ \$500) and user-friendly architecture
  - Manufacturing technology agnostic (e.g. 3D printing, laser cutting, etc.) designs
  - Modularly reconfigurable components/sub-systems
  - Integrated and comprehensive resources and tools
- Perception sub-system shall offer:
  - Ranging measurements (360° FOV, $\leq$ 1° resolution)
  - RGB visual feed (preferably front as well as rear)
  - Positional coordinates $\{X, Y, Z\}$ (error $\leq$ 5e-2 m)
  - Inertial measurements $\{a_x, a_y, a_z, \omega_x, \omega_y, \omega_z\}$ and AHRS estimates $\{\phi_x, \theta_y, \psi_z\}$
  - Actuation feedback measurements $\{\tau, \delta\}$
- Computation and communication sub-systems shall offer:
  - Hierarchical computation topology (high/low-level)
  - GPU-enabled high-level edge computation platform
  - Embedded low-level computation platform
  - Vehicle-to-everything (V2X) communication
- Locomotion and signaling sub-systems shall offer:
  - Kinodynamically constrained drivetrain and steering
  - Standard automotive lighting and signaling

The functional system requirements were decomposed into mechanical, electronics, firmware and ADSS design specifications and carefully studied to analyze any potential trade-offs so as to finalize the components and ultimately come up with a refined system architecture design (refer Fig. 3).

The proposed hardware-software architecture of the scaled autonomous vehicle system is divided into eight sub-systems viz. chassis, power, computation, communication, software, sensors, actuators and lights, each with its own share of components (refer Fig. 3-A). The embedded firmware architecture for low-level data acquisition and control is depicted in Fig.

3-B, which links the data sources to the respective sinks after processing the information. Finally, Fig. 3-C depicts high-level architecture of the autonomous parking solution described in this paper. Particularly, it is shown how this candidate autonomy solution uses modular algorithms for simultaneous localization and mapping (SLAM) [25], odometry estimation [26], localization [27], global [28] and local [29] path planning, and motion control. Implementation descriptions are necessarily brief due to the space limitations; however, further details can be found in this technical report [23].

III. VIRTUAL PROTOTYPING AND TESTING

Virtual prototypes help expedite the design process by validating the designs against system requirements through simulation, and suggesting design revisions at an early stage. Additionally, they help reduce testing costs, improve safety and facilitate collaboration among design teams. Furthermore, virtual prototypes may be the only viable option for conducting tests such as corner and edge case analysis, variability and repeatability analysis, robustness and stress testing in high-risk conditions, to name a few.

The scaled autonomous vehicle system was virtually prototyped and tested in three phases. First, the mechanical specifications, motions and fit were carefully analyzed using a parametric computer aided design (CAD) assembly of the system in conjunction with the physical modeling approach for multi-body dynamic systems (refer Fig. 4-A). Parallelly, the electronic sub-systems were prototyped using the physical modeling approach, and also by adopting electronic design automation (EDA) workflow (refer Fig. 4-B). Next, the firmware for low-level control (front wheel steering angle and rear wheel velocity) of the vehicle was verified to produce reliable results (within a specified tolerance of 3e-2 rad for steering angle and 3e-1 rad/s for wheel velocity) through model-in-the-loop (MIL) and software-in-the-loop (SIL) testing (refer Fig. 4-C).

The knowledge gained through this process was used to update the AutoDRIVE Simulator (refer Fig. 4-D) from its
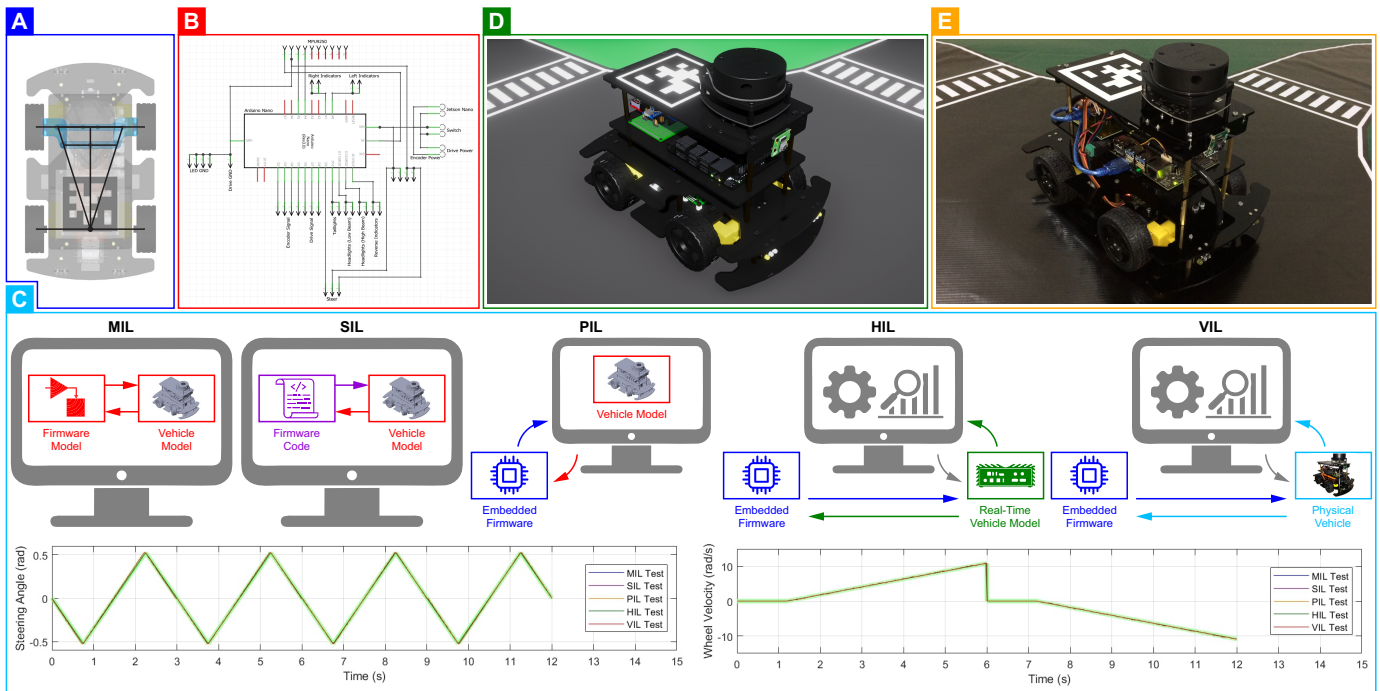
Fig. 4. Development and system integration of scaled autonomous vehicle: [A] mechanical assembly; [B] electronic schematic; [C] MBD workflow depicting MIL, SIL, PIL, HIL and VIL testing of vehicle firmware; [D] virtual prototype in AutoDRIVE Simulator; [E] physical prototype in AutoDRIVE Testbed.

initial version discussed in [30], [31] to the one described in [22], [23]. The updated simulator was then employed for verification and validation of individual ADSS modules and finally, the integrated autonomous parking solution was also verified using the same toolchain (refer Fig. 5-A). Particularly, we tested the vehicle in multiple environments, which included unit tests for validating the SLAM, odometry, localization, planning and control algorithms, followed by verification of the integrated pipeline with and without the addition of dynamic obstacles, which were absent while mapping the environment. The autonomous navigation behavior was analyzed for 5 trials and verified to fit within an acceptable tolerance threshold of 2.5e-2 m; the acceptable parking pose tolerance was set to be 5e-2 m for linear positions in X and Y directions and 8.73e-2 rad for the angular orientation about Z-axis.

IV. Hybrid Prototyping and Testing

All models or virtual prototypes involve certain degrees of abstraction, ranging from model fidelity to simulation settings, and as such, cannot be treated as perfect representations of their real-world counterparts. Therefore, once the virtual prototyping and preliminary testing of the system has been accomplished, the next step is to prototype and validate it in a hybrid fashion (partly virtual and partly physical), focusing more on high-level system integration. This method of hybrid prototyping and testing is extremely beneficial since it follows a gradual transition from simulation to reality, thereby enabling a more faithful system verification framework and providing a room for potential design revisions even before complete physical prototyping is accomplished.

The scaled vehicle system was subjected to hybrid testing by running processor-in-the-loop (PIL), hardware-in-the-loop (HIL) and vehicle-in-the-loop (VIL) tests on the embedded firmware for confirming minimum deviation from MIL and SIL results, specified by the same tolerance values of 3e-2 rad for steering angle and 3e-1 rad/s for wheel velocity (refer Fig. 4-C). The performance of integrated autonomous vehicle system was then validated using hybrid testing in two phases.

First, we deployed the ADSS on the physical vehicle's on-board computer, which was interfaced with AutoDRIVE Simulator to receive live sensor feed from the virtual vehicle, process it and generate appropriate control commands, and finally relay these commands back to the simulated vehicle. Specifically, for the autonomous parking solution (refer Fig. 5-A), we deployed and tested each of the SLAM, odometry, localization, planning and control algorithms for satisfactory performance. This was naturally followed by deployment and validation of the integrated pipeline for accomplishing reliable (within a precision tolerance of 2.5e-2 m) source-to-goal navigation (within a goal pose tolerance of 5e-2 m and 8.73e-2 rad) in different environments, wherein a subset of cases included dynamic obstacles as discussed earlier.

Next, we collected real-world sensor data using AutoDRIVE Testbed and replayed it as a real-time stimulus to the ADSS deployed on the physical vehicle's on-board computer running in-the-loop with AutoDRIVE Simulator. This way, we increased the "real-world" component of the hybrid test and verified the autonomous parking solution for expected performance (within same tolerance values as mentioned earlier). Particularly, the real-world data being collected/replayed was
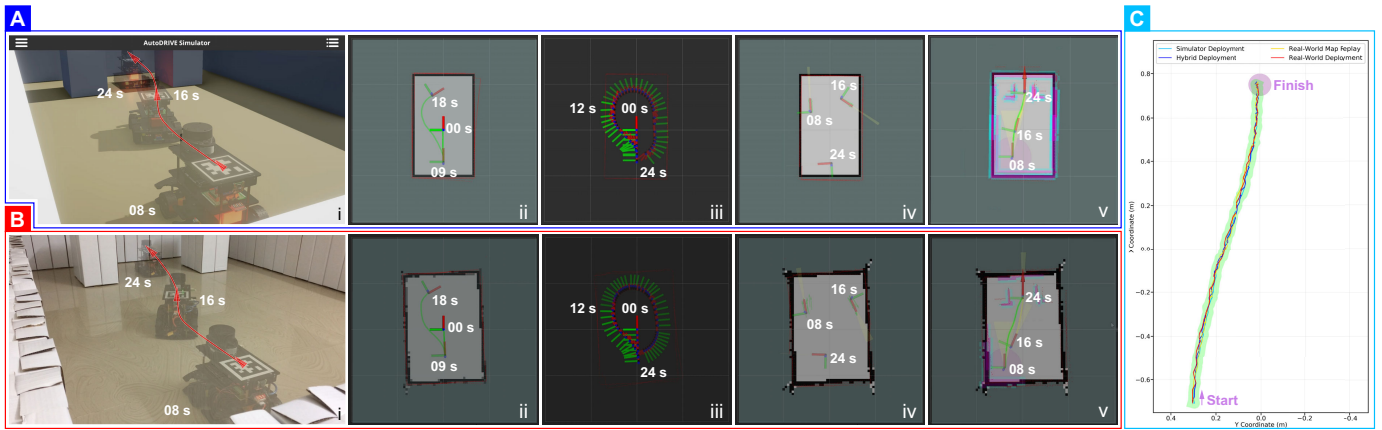
Fig. 5. Verification and validation of scaled autonomous vehicle performance: [A] virtual/hybrid and [B] physical validation of (i) integrated system, unit testing of (ii) SLAM, (iii) odometry, (iv) localization, (v) planning and control modules in AutoDRIVE Simulator/Testbed; [C] repeatability/reliability analysis represented as mean and standard deviation of 5 trials for each deployment type with acceptable trajectory tolerance in green and parking tolerance in purple.

occupancy-grid map of the environment built by executing the SLAM module on the physical vehicle, which inherently resulted as a unit test of this module in real-world conditions. The simulated vehicle had to then localize against this real-world map while driving in the virtual scene and navigate autonomously from source to goal (parking) pose, which further tested the robustness of the integrated pipeline against minor environmental variations and/or vehicle behavior.

V. Physical Prototyping and Testing

Verification and validation of safety-critical systems such as autonomous vehicles is quite difficult owing to their inherent complexity. Although virtual and hybrid testing approaches expedite the initial design and development process, these systems need to be ultimately tested in proving-ground or real-world conditions to validate their performance. Additionally, the question of simulation fidelity and benchmarking of virtual prototypes against their physical counterparts must be addressed. Consequently, once these systems confirm satisfactory performance under hybrid testing conditions, the next and final stage in mechatronic system development is physical prototyping and testing (refer Fig. 4-E).

In order to physically validate the autonomous vehicle (refer Fig. 5-B), we initially carried out unit tests to confirm the performance of each of the individulal algorithms for SLAM, odometry, localization, planning and control, followed by deployment of the integrated software stack for autonomous parking application (refer Fig. 5-C). The vehicle was confirmed to exhibit a reliable (within a precision tolerance of 2.5e-2 m) source-to-goal navigation (within a goal pose tolerance of 5e-2 m and 8.73e-2 rad). Once again, as in case of virtual prototyping, the robustness of ADSS was tested by introducing dynamic obstacles that were not present while environment mapping was performed. This physical experimentation also helped benchmark AutoDRIVE Simulator against AutoDRIVE Testbed; that is to say, it was confirmed that the virtual and physical prototypes exploited for validating the aforementioned candidate autonomy algorithm have high resemblance in terms of physical (kinematics and dynamics) and graphical (dimensional accuracy and photorealistic rendering) fidelity of the vehicle as well as environment.

VI. Conclusion

The key contributions of this work include: (i) formal extension of the V-model describing mechatronics approach of system design, verification and validation for autonomous vehicles; (ii) adoption of AutoDRIVE Ecosystem for modular and reconfigurable multidisciplinary design of autonomous vehicles across scales; and (iii) setting up a streamlined workflow for virtual, hybrid as well as physical prototyping and testing of autonomous vehicles in the context of an autonomous parking application case study.

Particularly, we presented an extended V-model fostering mechatronics approach of system design, verification and validation for autonomous vehicles. Further, we discussed how AutoDRIVE Ecosystem aims to adopt and promote the mechatronics approach for autonomous vehicle development across scales and inculcate a habit of following it from academic education and research to industrial deployments. We also demonstrated the methodical adoption of mechatronics approach for designing, developing and validating a scaled autonomous vehicle in the context of a detailed case study pertaining to autonomous parking using a modular probabilistic framework; including both qualitative and quantitative remarks. We showed that the design, development as well as verification and validation of the scaled autonomous vehicle with regard to the aforementioned case study could be successfully accomplished within a stringent time-frame of about one month [23]. It is to be noted that although the exact timeline of any multidisciplinary project may vary depending upon factors such as skill set, experience and number of individuals involved, lead time in the supply chain, etc., the mechatronics approach definitely proves to be efficient in terms of minimizing the design-development iterations by the virtue of synergistic integration in a concurrent engineering thinking framework. This provides a room for the rectification

of any design issues early in the development cycle, thereby increasing the chances of successful verification and validation with minimal loss of time and resources.

In order to better compare the mechatronics approach against others (in the context of autonomous vehicles), we propose some qualitative (degree of integration, flexibility, functionality, safety, reliability, etc.) and quantitative (development time, cost, performance, efficiency, complexity, etc.) metrics, which we plan to investigate in a future work.

REFERENCES

[1] S. Beiker, F. Hansson, A. Suneson, and M. Uhl, "How the convergence of automotive and tech will create a new ecosystem," *McKinsey Company*, 2016.

[2] F. Meissner, K. Shirokinskiy, and M. Alexander, "Computer on Wheels: Disruption in Automotive Electronics and Semiconductors," *Roland Berger: Munich, Germany*, 2020.

[3] J. A. Gumiel, J. Mabe, J. Jiménez, and J. Barruetabeña, "Introducing the Electronic Knowledge Framework into the Traditional Automotive Suppliers' Industry: From Mechanical Engineering to Mechatronics," *Businesses*, vol. 2, no. 2, pp. 273–289, 2022. [Online]. Available: https://www.mdpi.com/2673-7116/2/2/18

[4] W. Bolton, *Mechatronics: Electronic Control Systems in Mechanical Engineering*. Longman Scientific & Technical, 1995. [Online]. Available: https://books.google.com/books?id=XiBTAAAAMAAJ

[5] C. de Silva, *Mechatronics: An Integrated Approach*. Taylor & Francis, 2004. [Online]. Available: https://books.google.com/books?id=CjB2ygeR95cC

[6] G. Onwubolu, *Mechatronics: Principles and Applications*. Elsevier Science, 2005. [Online]. Available: https://books.google.com/books?id=bgK8kCMpD_YC

[7] J. Gausemeier and S. Moehringer, "VDI 2206- A New Guideline for the Design of Mechatronic Systems," *IFAC Proceedings Volumes*, vol. 35, no. 2, pp. 785–790, 2002, 2nd IFAC Conference on Mechatronic Systems, Berkeley, CA, USA, 9-11 December. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1474667017340351

[8] A. Bröhl and W. Dröschel, *Das V-Modell: der Standard für die Softwareentwicklung mit Praxisleitfaden*, ser. Software - Anwendungsentwicklung - Informationssysteme. Oldenbourg, 1995. [Online]. Available: https://books.google.com/books?id=B2iKAAAACAAJ

[9] M. Eigner, T. Dickopf, and H. Apostolov, "The Evolution of the V-Model: From VDI 2206 to a System Engineering Based Approach for Developing Cybertronic Systems," in *Product Lifecycle Management and the Industry of the Future*, J. Ríos, A. Bernard, A. Bouras, and S. Foufou, Eds. Cham: Springer International Publishing, 2017, pp. 382–393. [Online]. Available: https://doi.org/10.1007/978-3-319-72905-3_34

[10] P. Pathrose, *ADAS and Automated Driving: A Practical Approach to Verification and Validation*. SAE International, Jun 2022. [Online]. Available: https://doi.org/10.4271/R-525

[11] *Road vehicles - Functional safety*, ISO 26262:2018 ed. Geneva, CH: International Organization for Standardization, 2018.

[12] *Information technology - Process assessment*, ISO/IEC TS 33061:2021 ed. Geneva, CH: International Organization for Standardization and International Electrotechnical Commission (ISO/IEC JTC 1), 2021.

[13] *Design of technical products and systems*, VDI 2221:2019 ed. Düsseldorf, DE: The Association of German Engineers, 2019.

[14] *Development of mechatronic and cyber-physical systems*, VDI 2206:2021 ed. Düsseldorf, DE: The Association of German Engineers, 2021.

[15] S. Fürst, J. Mössinger, S. Bunzel, T. Weber, F. Kirschke-Biller, P. Heitkämper, G. Kinkelin, K. Nishikawa, and K. Lange, "AUTOSAR - A Worldwide Standard is on the Road," in *14th International VDI Congress Electronic Systems for Vehicles, Baden-Baden*, vol. 62, 2009, p. 5.

[16] S. Karaman, A. Anders, M. Boulet, J. Connor, K. Gregson, W. Guerra, O. Guldner, M. Mohamoud, B. Plancher, R. Shin, and J. Vivilecchia, "Project-based, collaborative, algorithmic robotics for high school students: Programming self-driving race cars at MIT," in *2017 IEEE Integrated STEM Education Conference (ISEC)*, 2017, pp. 195–203. [Online]. Available: https://mit-racecar.github.io

[17] M. O'Kelly, V. Sukhil, H. Abbas, J. Harkins, C. Kao, Y. V. Pant, R. Mangharam, D. Agarwal, M. Behl, P. Burgio, and M. Bertogna, "F1/10: An Open-Source Autonomous Cyber-Physical Platform," 2019. [Online]. Available: https://arxiv.org/abs/1901.08567

[18] S. S. Srinivasa, P. Lancaster, J. Michalove, M. Schmittle, C. Summers, M. Rockett, J. R. Smith, S. Choudhury, C. Mavrogiannis, and F. Sadeghi, "MuSHR: A Low-Cost, Open-Source Robotic Racecar for Education and Research," 2019. [Online]. Available: https://arxiv.org/abs/1908.08031

[19] N. P. Koenig and A. Howard, "Design and use paradigms for Gazebo, an open-source multi-robot simulator," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, vol. 3, 2004, pp. 2149–2154 vol.3.

[20] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An Open Urban Driving Simulator," in *Proceedings of the 1st Annual Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, S. Levine, V. Vanhoucke, and K. Goldberg, Eds., vol. 78. PMLR, 13-15 Nov 2017, pp. 1–16. [Online]. Available: http://proceedings.mlr.press/v78/dosovitskiy17a.html

[21] Cognata, "Cognata," Sep 2022. [Online]. Available: https://www.cognata.com/

[22] T. V. Samak, C. V. Samak, S. Kandhasamy, V. Krovi, and M. Xie, "AutoDRIVE: A Comprehensive, Flexible and Integrated Cyber-Physical Ecosystem for Enhancing Autonomous Driving Research and Education," 2022. [Online]. Available: https://arxiv.org/abs/2212.05241

[23] T. V. Samak and C. V. Samak, "AutoDRIVE - Technical Report," 2022. [Online]. Available: https://arxiv.org/abs/2211.08475

[24] W. Young, W. Boebert, and R. Kain, "Proving a computer system secure," *The Scientific Honeyweller*, vol. 6, no. 2, pp. 18–27, 1985.

[25] S. Kohlbrecher, O. von Stryk, J. Meyer, and U. Klingauf, "A Flexible and Scalable SLAM System with Full 3D Motion Estimation," in *2011 IEEE International Symposium on Safety, Security, and Rescue Robotics*, 2011, pp. 155–160. [Online]. Available: http://doi.org/10.1109/SSRR.2011.6106777

[26] M. Jaimez, J. G. Monroy, and J. Gonzalez-Jimenez, "Planar Odometry from a Radial Laser Scanner. A Range Flow-based Approach," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 4479–4485. [Online]. Available: http://doi.org/10.1109/ICRA.2016.7487647

[27] D. Fox, "KLD-Sampling: Adaptive Particle Filters," in *Advances in Neural Information Processing Systems*, T. Dietterich, S. Becker, and Z. Ghahramani, Eds., vol. 14. MIT Press, 2001. [Online]. Available: https://proceedings.neurips.cc/paper/2001/file/c5b2cebf15b205503560c4e8e6d1ea78-Paper.pdf

[28] P. E. Hart, N. J. Nilsson, and B. Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," *IEEE Transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968. [Online]. Available: http://doi.org/10.1109/TSSC.1968.300136

[29] C. Rösmann, F. Hoffmann, and T. Bertram, "Kinodynamic Trajectory Optimization and Control for Car-Like Robots," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 5681–5686. [Online]. Available: http://doi.org/10.1109/IROS.2017.8206458

[30] T. V. Samak, C. V. Samak, and M. Xie, "AutoDRIVE Simulator: A Simulator for Scaled Autonomous Vehicle Research and Education," in *2021 2nd International Conference on Control, Robotics and Intelligent System*, ser. CCRIS'21. New York, NY, USA: Association for Computing Machinery, 2021, p. 1–5. [Online]. Available: https://doi.org/10.1145/3483845.3483846

[31] T. V. Samak and C. V. Samak, "AutoDRIVE Simulator - Technical Report," 2022. [Online]. Available: https://arxiv.org/abs/2211.07022