Modeling, Simulation and Visualization Student Capstone Conference

Apr 14th, 12:00 AM - 12:00 AM

Synthesizing Maritime Interaction Scenarios for Testing Autonomy

Benjamin E. Hargis
Old Dominion University

Yiannis E. Papelis
Old Dominion University

Recommended Citation

Hargis, Benjamin E. and Papelis, Yiannis E., "Synthesizing Maritime Interaction Scenarios for Testing Autonomy" (2022). Modeling, Simulation and Visualization Student Capstone Conference. 1. DOI: 10.25776/52yt-5v43 https://digitalcommons.odu.edu/msvcapstone/2022/scienceengineering/1

SYNTHESIZING MARITIME INTERACTION SCENARIOS FOR TESTING AUTONOMY

Benjamin E. Hargis and Yiannis E. Papelis
Department of Computational Modeling and Simulation Engineering
Old Dominion University
5115 Hampton Boulevard Norfolk, VA, USA
bharg010@odu.edu

ABSTRACT

This paper presents a method to deterministically synthesize maritime traffic interactions that can be presented to a system under test regardless of the state of the system under test. A background to the problem is given and the method is briefly outlined. Results indicate that the approach can enable more robust evaluation of maritime autonomous algorithms.

Keywords: Autonomy Algorithm Evaluation, Simulation, Testing

1 INTRODUCTION

As the desire to equip Autonomous Surface Vessels with increasingly complex autonomy grows, development of effective testing methodologies for autonomy-based behaviors is of importance. Traditional simulation-based single-event testing, however, lacks the ability to evaluate the long-term performance of autonomy algorithms before in-situ testing takes place. One approach is the use of long-run simulations which more accurately represents the intended operating timelines. An issue with designing events in long-run simulations with multiple consecutive events is that the state of the system under test (SUT) is not known before runtime. Therefore, methods for designing interactions that are specific, yet versatile and flexible are required. This paper presents a method to deterministically synthesize maritime traffic interactions, performed by a simulation entity referenced as an intruder from here forward, that can be presented to a system under test regardless of the state (i.e. position, velocity, etc.) of the SUT. The development was motivated by three factors. First, the developed method needs to allow a targeted interaction design. This allows the continued use of recognized approaches to scenario generate scenario description, knowledge driven and data driven (Nalic et al. 2020). Second, the method should accommodate closed-feedback testing approaches that can select testing situations based on prior performance of the system under test. Due to solution space dimensionality, exhaustive scenario generation is intractable (Koopman et al. 2016). Accommodating closed feedback enables solution space reduction methods such as performance boundary identification (Mullins et al. 2017, Stakiewicz et al. 2019). Finally, the method must facilitate testing time compression by reducing and/or eliminating time during the simulation when the SUT is not being stressed by external factors. Data-driven scenarios are useful for providing actual encounters that were observed yet encounters for a single mission represent only a small percentage of the mission time and methods to use only those portions of the mission data are being developed (Refai et al. 2019). Therefore, represent data-driven scenarios using the developed method would allow for relevant scenario based testing while still providing time compression.

2 EVENT SYNTHESIS METHOD

The basis of this approach is the application of Relative-Motion principles from Dynamics. These principles allow for event synthesis to be conducted in the SUTs local frame and then mapped or transformed to the world frame such that, from the perspective of the SUT, the presentation of event will be the same regardless of the SUT state. The event timeline has been discretized, by time, into three phases. The first phase, *Orchestration*, begins at time t_0 which marks the beginning of the event synthesis process. The second phase, *Implementation*, marks the end of the orchestration phase at the start time, t_s, and indicates the time when the intruder, INT, begins the desired event trajectory. The final phase, post-event, begins following the event time, t_e. Although the interaction between the intruder and the SUT has already taken place, this phase will define the intruder's actions after the assigned event. The orchestration phase uses the distance at closest point of approach (DCPA), time of event t_e, trajectory start time t_s, intruder velocity, system under SUT state, and heading difference δ, to determine the trajectory of the intruder. First, predict SUT state assuming constant heading and velocity. Then determine the intruder's pose (position and orientation) with respect to the SUT at t_e and t_s using the design parameters and constant heading and velocity assumptions. These thus define the trajectory when coupled with the respective times (t_e and t_s). At time t_s, the intruder begins the trajectory. As the SUT and intruder evolves through time, SUT referenced intruder poses are transformed into world space poses. This persists until the event occurs at t_e. The *Post-event* phase occurs following the event and allows definition of intruder actions. For this paper, the intruder is assumed to maintain heading and velocity after t_e.

3 CONCLUSIONS AND FUTURE WORK

The method was validated using a continuous simulation built in MATLAB 2021b. Two additional models were needed for implementation to include an environment model (a frictionless, massless, plane located in a vacuum) and a SUT model (a point mass with motion described using rigid body kinematics). The simulation evolved the appropriate states through time using Euler Integration. The results validate the use of this method to synthesize varying maritime traffic interactions by comparing input parameters to performance data. Future work could include the development of the complexity parameter along with an investigation of measurement noise on the calculated trajectory. For example, one method to increase the event complexity is by defining a percentage of time between t_s and t_e that the intruder will actively pursue the desired event parameters.

REFERENCES

D. Nalic, T. Mihalj, M. Bäumler, M. Lehmann, and S. Bernsteiner, "Scenario Based Testing of Automated Driving Systems: A Literature Survey," FISITA Web Contress, 2020. Accessed on: March. 11, 2022. URL: https://www.researchgate.net/publication/344896382

P. Koopman and M. Wagner, "Challenges in Autonomous Vehicle Testing and Validation," *SAE Int. J. Trans. Safety*, vol. 4, no. 1, pp. 15–24, Apr. 2016, doi: 10.4271/2016-01-0128.

G. E. Mullins, P. G. Stankiewicz, R. C. Hawthorne, and J. D. Appler, "Delivering Test and Evaluation Tools for Autonomous Unmanned Vehicles to the Fleet," Johns Hopkins APL Technical Digest, vol. 33, no. 4, p. 10, 2017.

P. G. Stankiewicz and G. E. Mullins, "Improving Evaluation Methodology for Autonomous Surface Vessel COLREGS Compliance," in OCEANS 2019 - Marseille, Marseille, France, Jun. 2019, pp. 1–7, doi: 10.1109/OCEANSE.2019.8867549.

M. Refai, M. Abramson, S. Lee, and G. Wu, "Encounter-Based Simulation Architecture for Detect-And-Avoid Modeling," presented at the AIAA Scitech 2019 Forum, San Diego, California, Jan. 2019. doi: 10.2514/6.2019-1476.