2018 NDIA GROUND VEHICLE SYSTEMS ENGINEERING AND TECHNOLOGY SYMPOSIUM
AUTONOMOUS GROUND SYSTEMS (AGS) TECHNICAL SESSION
AUGUST 7-9, 2018 - NOVI, MICHIGAN

DEVELOPING AUTONOMOUS STREET-LEGAL VEHICLES: ANALYSIS OF 2017 INTELLIGENT GROUND VEHICLE COMPETITION (IGVC) SELF-DRIVE/AUTO-NAV COMPETITION VEHICLE DESIGN AND IMPLEMENTATION

Andrew Kosinski
Mechanical Engineer
US Army TARDEC
Warren, MI

Jane Tarakhovsky
Self-Drive Chair
Hyundai Mobis
Plymouth, MI

Kiran Iyengar (Electrical Engineer, US Army TARDEC)
Jerry Lane (GLS&T, IGVC Co-Founder)
KaC Cheok (Oakland University, IGVC Co-Founder)
Bernie Theisen (US Army TARDEC, IGVC Co-Chair)

ABSTRACT

The IGVC offers a design experience that is at the very cutting edge of engineering education. It is multidisciplinary, theory-based, hands-on, team implemented, outcome assessed, and based on product realization. It encompasses the very latest technologies impacting industrial development and taps subjects of high interest to students. Design and construction of an Intelligent Vehicle fits well in a two semester senior year design capstone course, or an extracurricular activity earning design credit. The deadline of an end-of-term competition is a real-world constraint that includes the excitement of potential winning recognition and financial gain. Students at all levels of undergraduate and graduate education can contribute to the team effort, and those at the lower levels benefit greatly from the experience and mentoring of those at higher levels. Team organization and leadership are practiced, and there are even roles for team members from business and engineering management, language and graphic arts, and public relations. Students solicit and interact with industrial sponsors who provide component hardware and advice, and in that way get an inside view of industrial design and opportunities for employment.

INTRODUCTION

Figure 1. IGVC 2017 Team Line-up.

The IGVC is a college level autonomous unmanned ground vehicle (UGV) competition that encompasses a wide variety of engineering professions – mechanical, electrical, computer engineering and computer science. It requires engineering students from these varied professions to collaborate in order to develop a truly integrated engineering product, a fully autonomous UGV.

The technologies involved in the IGVC come from a wide range of disciplines and are those of great current interest in both industry and engineering education. The technologies involved in the IGVC are those of emerging and burgeoning industries today. Among those applications are many with great opportunities for breakthroughs and innovation, and employment opportunities for knowledgeable young engineers abound.

There are four competitions within IGVC, the Design Competition, Auto-Nav Challenge, Interoperability Profile (IOP) Challenge and Self-Drive Challenge.

The Design Competition challenges students to document their vehicle development by creating a design report, followed by an in-person presentation to the design judges during the actual IGVC event, including a vehicle examination by the judges.

The Auto-Nav Challenge is the main challenge, which consists of an outdoor obstacle course that requires the UGVs to perform full autonomous operation/navigation throughout. The course is approximately 600 feet long in an area 100ft wide and 200 feet deep. Competitors can encounter natural or artificial inclines (ramps) with gradients not to exceed 15% and randomly placed obstacles along the course. Obstacles on the course consist of various colors (white, orange, brown, green, black, etc.) of construction barrels/drums that are used on roadways and highways.

The IOP Challenge encourages students to make their vehicles more interoperable, by requiring development of a Joint Architecture for Unmanned Systems (JAUS) compliant UGV, which is the architecture current military robots are being designed to. Programs such as the Robotic Operating System (ROS) are used by teams for designing/implementing software code, allowing for easier integration of new sensors and to help ensure commonality among the UGVs.

Figure 2. 2017 Self-Drive Challenge teams (Lawrence Technological University, Bob Jones University and Oakland University).

The Self-Drive Challenge is in its second year, requiring vehicles to perform road operations including lane keeping, lane switch, merging, avoiding crossing obstacles (simulated pedestrians/vehicles), taxi pickup of passengers, simulated pothole detection, stop and crosswalk lines detection, right/left turn and intersection detection/logic, navigation to GPS waypoints and autonomous parking.

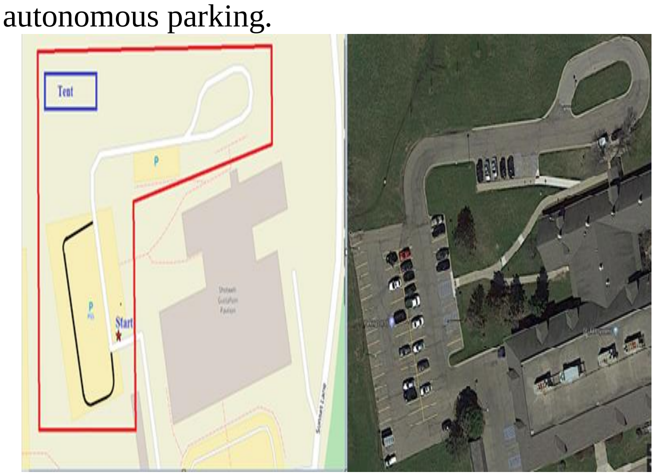


Figure 3. 2018 Self-Drive Challenge course (Oakland University Incubator, 419 Golf View Ln, Rochester, MI 48309).

Section 1.1. IGVC Self-Drive Challenge Design Specifications/Rules

The Self-Drive Challenge is a unique challenge vs the traditional Auto-Nav Challenge, as it is focused more on computer engineering/computer science challenges and software development, as most teams use a pre-existing COTS vehicle base (Polaris GEM e2, Renault Twizy, etc.), which can also include a pre-installed by-wire kit, resulting in the main remaining challenge being optimal software design/implementation and refined algorithms for best utilizing the data being generated from the various sensors/data components, including LIDARs, RADARs, cameras, GPS, IMU, etc. Computer engineering/computer science challenges include processing the raw sensor feeds, extracting relevant data, fusing the sensor feeds together and then utilizing this combined sensor data to plot optimal vehicle paths, avoiding obstacles, making correct vehicle behaviors during navigation (road sign detection/behavior, pedestrian/obstacle behavior, pothole detection, right/left turns, intersection detection/behaviors, parking maneuvers (pull in/pull out/parallel), merging, etc.).

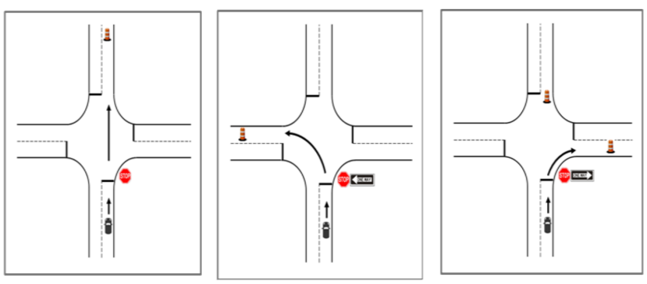


Figure 4. Lane keeping, left turn and right turn Self-Drive Challenge scenarios.[5]

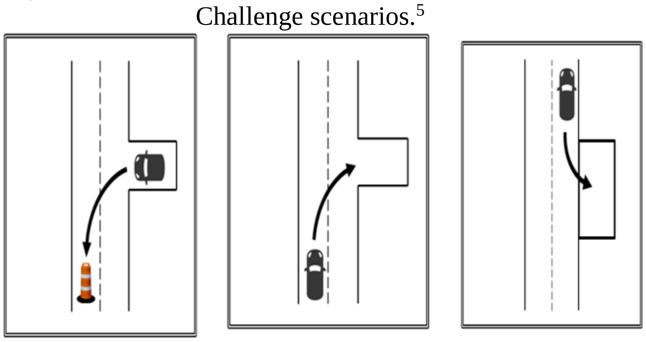


Figure 5. Pull-out, pull-in and parallel parking Self-Drive Challenge scenarios.[5]

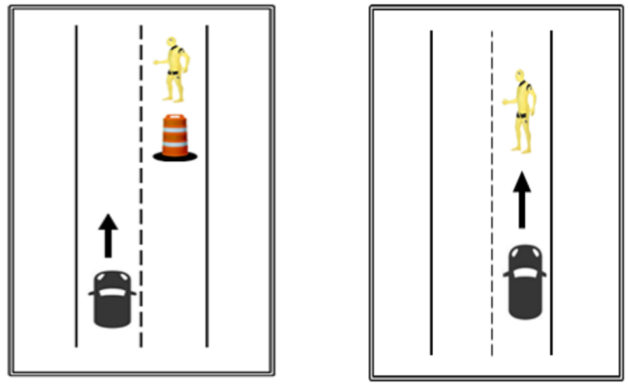


Figure 6. Obstructed/unobstructed pedestrian detection Self-Drive Challenge scenarios.[5]

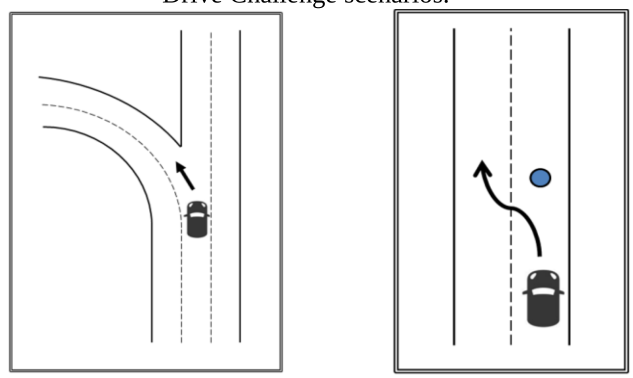


Figure 7. Merging and Pothole detection Self-Drive Challenge scenarios.[5]

Section 1.2. Lawrence Technological University's 2017 IGVC 1st Place Self-Drive Vehicle

Figure 8. Lawrence Technological University (LTU) Self-Drive vehicle running on LTU roadway.[6]

Lawrence Technological University, the 2017 Self-Drive Challenge winner, has been refining and applying their street-legal Self-Drive vehicle on their Southfield University campus with the goal of using it as an autonomous taxi service by Fall 2018. Due to the relevance/focus of the Self-Drive Challenge to the existing, growing interest by companies in autonomous street-legal vehicles, new IGVC sponsors have expressed interest in 2018, including OpenJAUS and Robotic Research, with several sponsors providing technical guidance to the various Self-Drive teams.

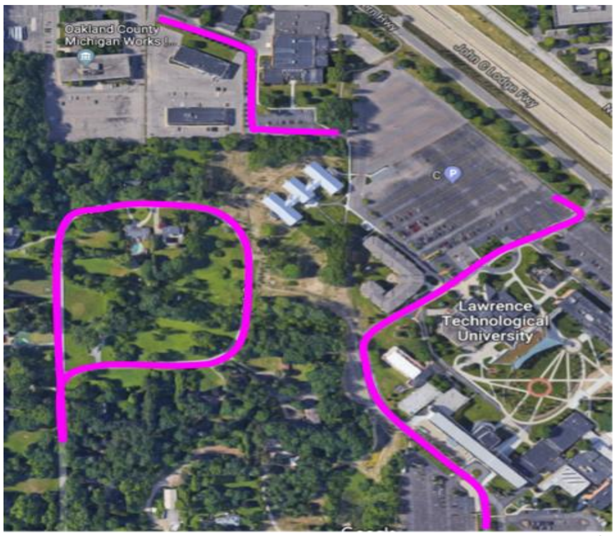


Figure 9. LTU Self-Drive vehicle roadway testing areas.[6]

As shown in Figure 9, LTU's Self-Drive vehicle has been run around a few different roadway locations within and nearby LTU, which include encounters with stop signs, lane markings, pedestrians and required compliance with rules of the road for standard street-legal vehicles. The LTU campus is an ideal test environment due to its low speed roads (25mph) and relatively constrained environment, with clear lane markings, traffic signs and designated crosswalk areas for pedestrians. Low speed roads have the benefit of greatly reduced chances of serious injury/death to the vehicle riders as well as pedestrians/vehicles nearby in case of a vehicle failure, as opposed to high speed as with the 2016/2018 Tesla fatal car crashes and the 2018 Uber fatal car crash, all involving varying levels of autonomy vehicles traveling at highway speeds.

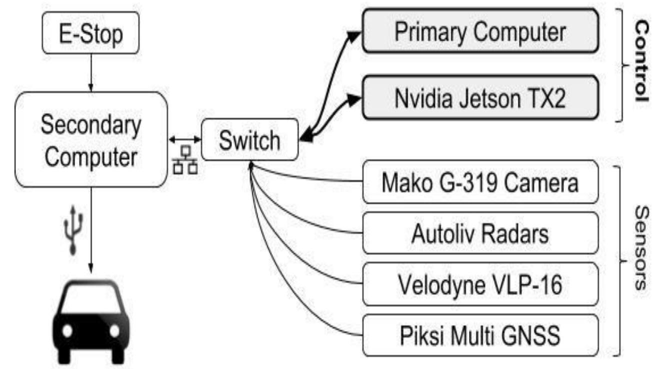


Figure 10. LTU Self-Drive vehicle safety/processing/sensor overview schematic.[6]

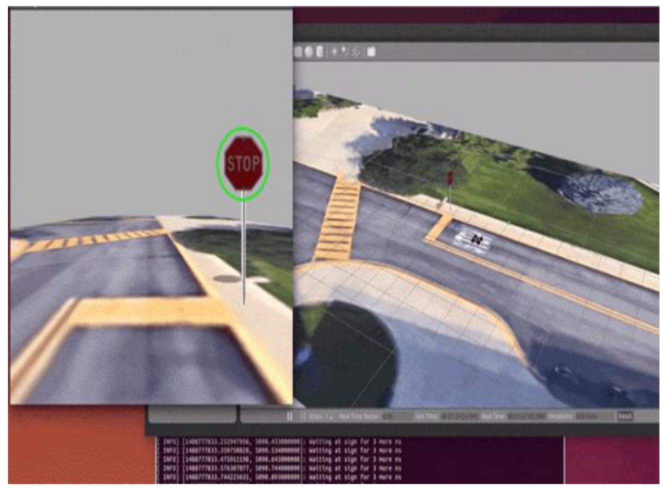


Figure 11. LTU Self-Drive vehicle simulation testing with relevant simulated environment (road signs, lane markings, etc.)[6]

Section 1.3. Smooth Control Path Planning Simulation

1. Mathematical Formulation

a. Definition of System Variables for Smooth Control Controller

Some of the important variables to keep in mind, as shown in Figure 14 are r, Θ, and δ. The distance from the robotic vehicle to its defined target is r, the orientation of the target with respect to the line of sight is Θ. The orientation of the robot heading with respect to the line of sight is δ to target, also shown in Figure 12. The linear velocity of the vehicle is given by v. The angular velocity of the vehicle is equal to ω. This control algorithm will be shown later in the paper to be implemented in a Leader-Follower simulation. Another way to apply this control model to a real world scenario is to consider that the robotic vehicle can represent a stopped car in a parking lot that is going to drive to an open space in the lot, which would represent a target. Therefore, it is helpful and optimal if the vehicle is lined up to the target to have the most ideal line of sight. When the vehicle reaches its target, the desired goal is to drive r and Θ to 0 [20, 24].

2. Lyapunov Stability Method for Smooth Control Law

The following equation can be defined as a Lyapunov candidate is also a positive definite function. This is shown by the following equation:

$$V = .5(r2 + \theta 2) > 0 \qquad 1.1$$

There should be a speed v and ω that produce a steering value δ that yields a distance r and an orientation Θ, so that the derivatives $\dot{r}$ and $\dot{\theta}$ result in the following equation.

$$\dot{V} = r\dot{r} + \Theta\dot{\Theta} \leq 0 \qquad 1.2$$

This results in the equation being a negative definite. The way to accomplish this is to find two values of the derivatives where both r and Θ approach 0. The Lyapunov Stability method states that a system is stable if a Lyapunov function v can be found where v>0 and $\dot{v} \leq 0$.

3. Smooth Control Law- Desire Vehicle Orientation

We can choose the following calculation as the desired orientation.

$$\delta = \tan\text{-}1\,(-k1\Theta) \qquad 1.3$$

Following this, we can obtain the resultant equation.

$$\dot{r} = -v\,(\cos(\tan\text{-}1(-k1\theta))) \qquad 1.4$$

Substituting the values in equation 1.2 results in equation 1.5 shown as the following

$$\dot{V} = r\dot{r} + \theta\dot{\theta} = -rv\cos((\tan\text{-}1(-k1\,\Theta))) + (v/r)\,\Theta \sin(\cos((\tan\text{-}1(-k1\Theta))) \leq 0 \qquad 1.5$$

The reasoning behind this is a result of the following equations.

$$\cos(\tan\text{-}1(-k1\theta)) > 0,\ \theta\epsilon(-\pi, \pi] \qquad 1.6$$

$$s\,(\tan-1\,(-k1)) = -sgn\,(\theta) \qquad 1.7$$

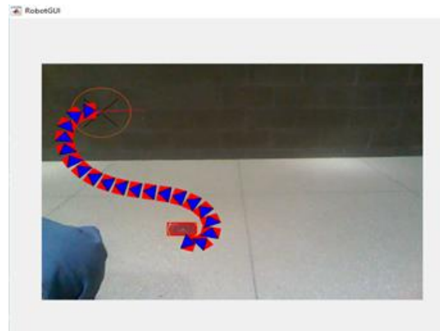


Figure 12. The path of the robot is shown by the simulation where the orientation is indicated by the arrows and the current position is shown.

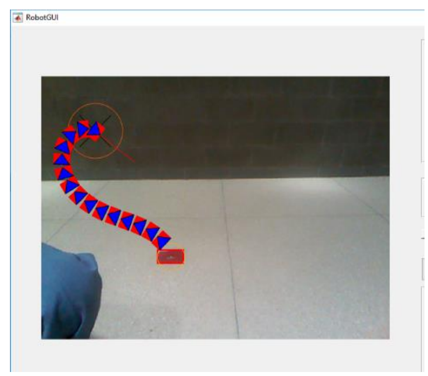


Figure 13. Here a different path is simulated path of the robot is shown where the position and orientation of the robot is different than Figure 12

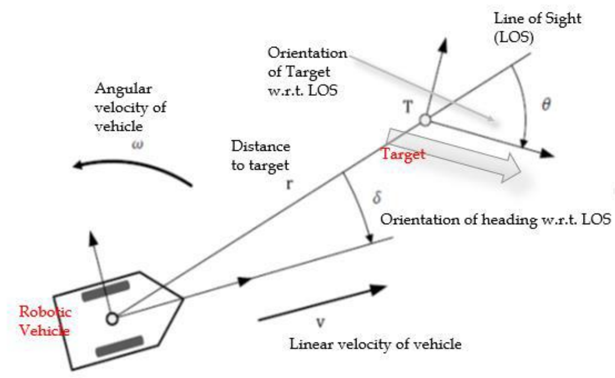


Figure 14. Top view diagram of the original physical system [20]

Section 1.4. Implementation of Smooth Control Law into Single Polaris Gem 2 for Open Field

1. MATLAB and Gazebo Implementation

The next step was to implement the control algorithm into a physical world model like Gazebo. This was done through a modified MATLAB script and the Robotics toolbox in MATLAB. Also, additional files in C++ were written to ensure ROS compatibility. Additionally, for testing purposes on one vehicle was used for this simulation to ensure that the algorithm would work. Furthermore, the physical visualization for the control scheme being implemented in a vehicle that works in open area as shown in the figure below [25].

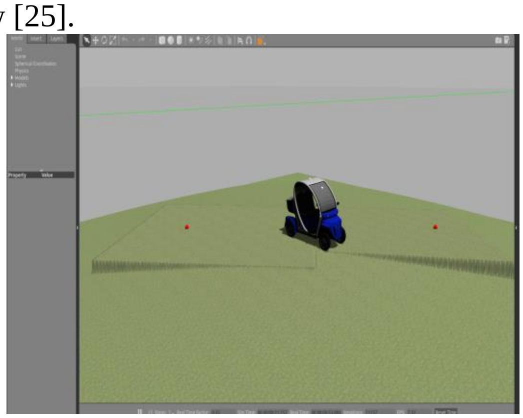


Figure 15. Here, a Polaris Gem 2 is shown in an open field in Gazebo in order to demonstrate that Smooth Control Law is implemented without any restrictions

Section 1.5. Leader-Follower Implementation for Multiple Polaris Gem 2 Vehicles

The next step for implementing the control scheme was to implement in a leader-follower scenario for the Self-Drive scenarios. This was done by further modification of MATLAB code and files in C++. However, as a result of course scenarios, restrictions had to be placed on the vehicles so that they would stay on course. The scenarios that the vehicles had to perform were lane keeping, stopping, left and right turn were implemented. Additionally, constant adjustments had to be made to ensure the vehicles adhered to the course rules. The following figures show the 3 vehicles on the course.

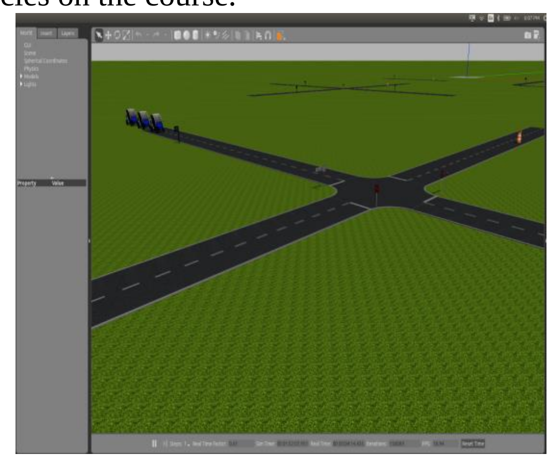


Figure 16. 3 Polaris Gem 2 Vehicles at starting position in a simulated function for Self-Drive course

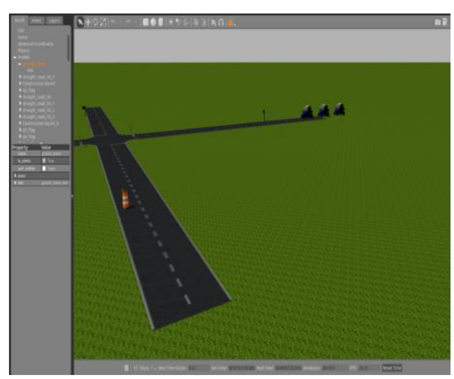


Figure 17. 3 Polaris Gem 2 Vehicles after completing a left turn

Section 1.6. Performance and Analysis of Simulation

The simulation performs as expected for various reasons. As shown by Figures 12 and 13, there are multiple paths that simulate the path of the robot using varying orientation and position of the robot depending on the specific point in the path. The orientation and position is adjusted throughout the path. Θ is varied throughout these paths. As a result, with a change in Θ, the path of the robot is varied multiple times. Also, the simulation shows the robot's adjustment along the path for having to continually implement the differential drive kinematics that were applied to it. An important contributing factor to achieving smooth path includes a significant reduction in noise of the vehicle. This method can help with a potential vehicle's turning ability and smooth path, as these results can be used in a similar simulation to help smooth out the turning of the vehicle after it is built and tested [20-24] target at a time. However, if another target is desired, then the real-time analysis must be run again. Additionally, multiple targets can be used in the case of a simulation. The smoothness of turns and path planning is also achieved when applying the mathematical equations from the simulation to functional motor control. Additionally, this is shown being implemented with ROS and Gazebo for 1 vehicle to ensure the control algorithm's functionality and then finally the 3 vehicles in the leader-follower scenario for Self-Drive to show that the scheme can be used for multiple functions as previously mentioned.

Section 2.1. Technical Challenge #1 – Frame/Suspension/Mast Selection/Design

Some of the engineering technical challenges mentioned above will now be explored in greater depth. Mechanical engineering challenges including designing/calculating appropriate placement of components to ensure an optimal vehicle center of gravity, as speed is a driving requirement for placing well in the Auto-Nav Challenge. UGV material selection must be performed, with teams tending to use aluminum for most frame components due to its low weight and ease of assembly using cheap, prefabricated aluminum components, such as 80/20 T-slotted aluminum framing.

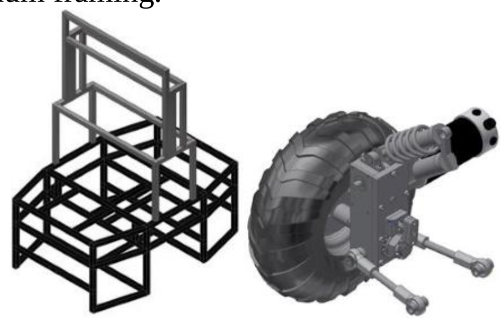


Figure 18. 2017 Georgia Institute of Technology 1"x1" square steel tubing using MIG welding and 1"x1" square aluminum tubing IGVC vehicle frame, and integrated motor-suspension system.[7]

Figure 19. 2017 Indian Institute of Technology Bombay's vehicle CAD model and frame views.[8]

This type of selection also has the benefit of allowing for easy swap out of components and simplified/quicker vehicle frame reconfiguration. An additional benefit to 80/20 T-slotted aluminum framing is that welding can be avoided. Optimal frame selection/mechanical design can be assisted through the use of CAD software and trade-off analysis.

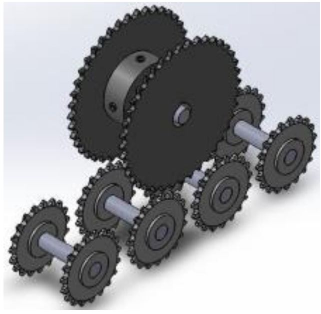
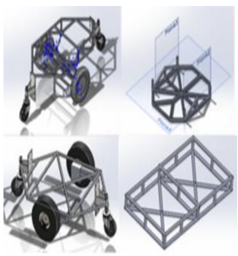


Figure 20. 2017 Oakland University's Octagon V3.0 tracked vehicle sketch/CAD design for enhanced off-road capabilities.[9]



	Ease of Fabrication	Weight	Stress Distribution	Score
Top Left	17	18	18	53
Top Right	13	16	14	43
Bottom Left	17	18	17	52
Bottom Right	20	17	18	55

Figure 21. 2017 Roger Williams University frame design down-select after FEA trade-off analysis.[10]

Suspension systems allowing for better component isolation to vibration/forces are usually included in the vehicles, typically centering on the use of traditional struts, springs, etc. Masts are usually installed for mounting of mono/stereo cameras to give a high point of view. Students usually use predrilled aluminum extrusion for easy/rapid adjustment of camera height.

Some teams use prefabricated frames/suspension systems, such as using electric wheelchairs. This has the advantage of reducing frame/suspension development time and testing, freeing up time for other IGVC vehicle work. Another benefit is taking advantage of a proven vehicle in terms of reliability, durability, etc. There is also increased team use of 3D printing to quickly fabricate parts and to simplify/speed up future design improvements/modifications. Modularization/compartmentalization of vehicle subcomponents also further simplifies future design alterations.

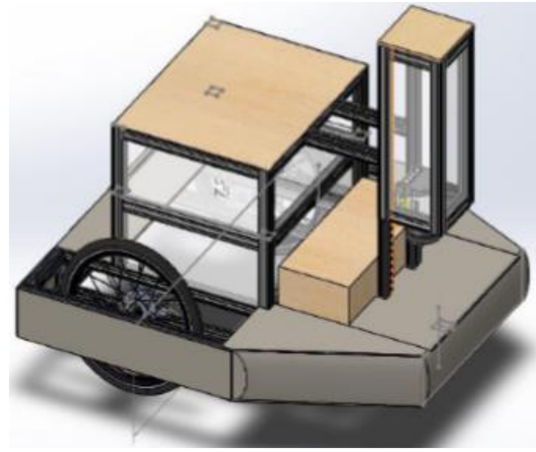


Figure 22. 2017 Indian Institute of Technology Madras, India incorporation of circuitry/power system compartmentalization and 3D printing for simplified vehicle modifications.[11]

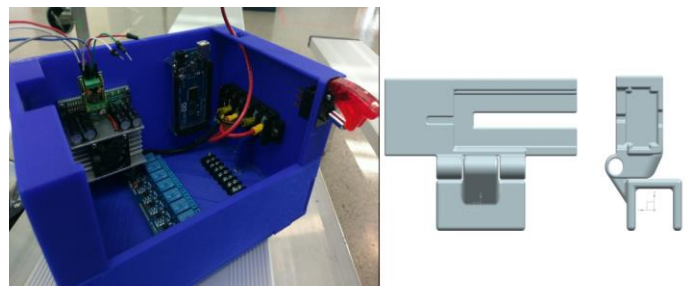


Figure 23. 2017 Lawrence Technological University's iWheels 3 3D printed electronics enclosure and camera mount.[12]

Section 2.2. Technical Challenge #2 – Vehicle Power/Battery/PCB Analysis/Selection/Fabrication

Motors must be selected, with students usually performing basic torque analysis to ensure the vehicle is capable of delivering an adequate overall vehicle speed, as well as being able to tackle the likely obstacles/environment the vehicle will face during the Auto-Nav completion, which as mentioned above can consist of slopes (up to 15° in Auto-Nav Challenge), potholes, muddy ground, grass/dirt, etc.

A typical torque analysis utilizes parameters such as vehicle weight, coefficient of friction, # of motors, wheel diameter, etc. Free-body diagrams can be used for necessary force calculations.

The various vehicle motors/sensors/circuit boards/components all have unique power draws at a variety of voltages, making power selection and distribution critical. Virtually all teams now use batteries as their vehicle's power source, although in the past fuel cells and combustion engines were used. Also, solar panels have been used by the US Naval Academy team in recent years to supplement battery power[4]. Battery technology has obviously advanced significantly over recent years, giving batteries a good form factor to power ratio. Lithium Ion and Lithium Polymer batteries are popular among teams, although lead acid are also still used.

Below is an example of the various common components requiring power in an IGVC vehicle, which is from the 2017 Hosei University design report:

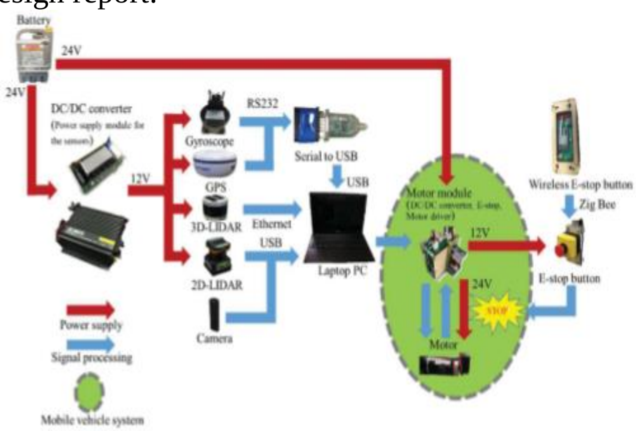
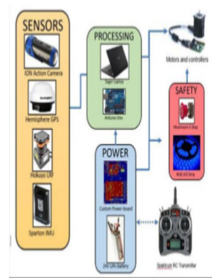


Figure 24. 2017 Hosei University's power/signal distribution diagram.[13]

As shown in the above pictures, and as mentioned partially in the beginning, common components include sensors (cameras and LADARs), PCBs, power converters/inverters, motor controllers, motors, e-stops and CPUs. Typical voltage requirements, as shown above, tend to be 5V, 12V and the motor's voltage. Clearly battery selection is of upmost importance, as if even one of these components receives too little power, the whole robot can become significantly degraded, if not entirely.

As mentioned above, battery selection is primarily dictated by overall power requirements of the various components, with typically the majority of the power draw coming from the drive motors. IGVC teams normally use wattmeters to determine the power draw of components at nominal and extreme load operations (extreme load usually equates to the vehicle operating at full desired course speed/incline).

Below are example tables of power draws of various IGVC vehicle subcomponents:



Sensor	Power Consumption	Voltage Range	Operating Voltage	Sources
Hemisphere A325 GPS	4.6 W	7 – 36 V	12 V	Power Board
Sparton GEDC-6E IMU	0.32 W	3.3 V	3.3 V	Laptop via USB
Ion Action Cam	1.5 W	3 – 5 V	3.7 V	Battery Pack
Hokuyo UTM-30LX-EW	8 W	10.8 – 13.2 V	12 V	Power Board
Quicksilver Motors	150 W	12 – 48 V	24 V	Power Board
E-Stop	0 W	10 – 40 V	12 V	Power Board
Laptop	6 W	19V	19V	Battery Pack

Figure 25. 2017 Embry-Riddle Aeronautical University's vehicle components power requirements.[14]

Electrical component	Max power consumption	Operating voltage	Source
Atlaslink GPS smart antenna	4.5 W	24 VDC	Li-ion battery pack
Sparton AHRS 8	0.875 W	5 VDC	Laptop via USB
SJCAM SJ40000 cameras	10 W	10 VDC	Laptop via USB
Ampflow A28-400 motors	750 W	24 VDC	Roboteq HDC
ASUS GR8 II mini-PC	230 W	19.5 VDC	DC-DC converter
Sick LMS111 LIDAR	8 W	24 VDC	Li-ion battery pack

Figure 26. 2017 Indian Institute of Technology Bombay's component power draws.[8]

From the data gathered from measuring power draws from the various components during the battery selection phase, necessary power distribution to the various components can be determined, normally then requiring the design of a printed circuit board (PCB). There are many CAD and PCB programs that can be used to create a virtual PCB, which can then be sent off for official manufacture.

Below is the virtual PCB side-by-side with the actual PCB created from this template for the 2017 Michigan Technological University/Oakland University vehicles:

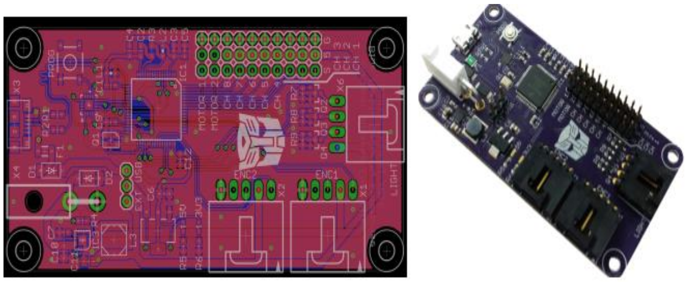


Figure 27. 2017 Michigan Technological University's virtual/actual PCB.[15]

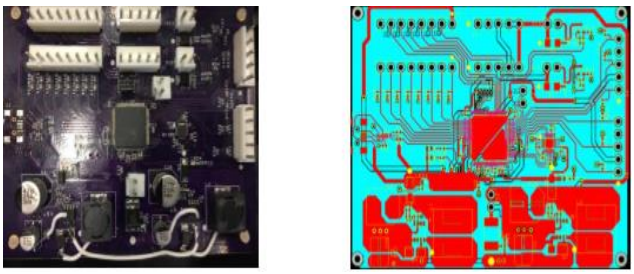


Figure 28. 2017 Oakland University Pinguino's virtual/actual PCB.[16]

Section 2.3. Technical Challenge #3 – Vehicle E-Stop/Safety Considerations/Design/Implementation

Emergency stops are a required component of every IGVC UGV. Without a physical e-stop on the vehicle and a wireless e-stop system (typical setup is a transmitter held by an IGVC field judge with an e-stop button on it, which typically triggers a vehicle circuit board with the singular function of safely powering down the UGV), the vehicle is not allowed to run the Auto-Nav Challenge. Normally the e-stop functions by killing power to the motor controllers.

Below are models/wiring diagrams/pictures of typical e-stop vehicle layouts/installation:

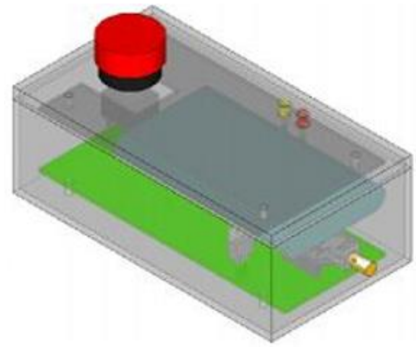


Figure 29. 2017 Rochester Institute of Technology e-stop 3-D model.[17]

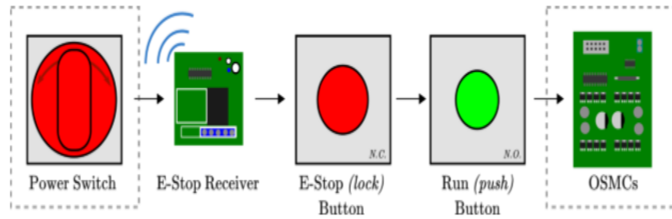


Figure 30. 2017 Georgia Institute of Technology e-stop vehicle setup.[7]

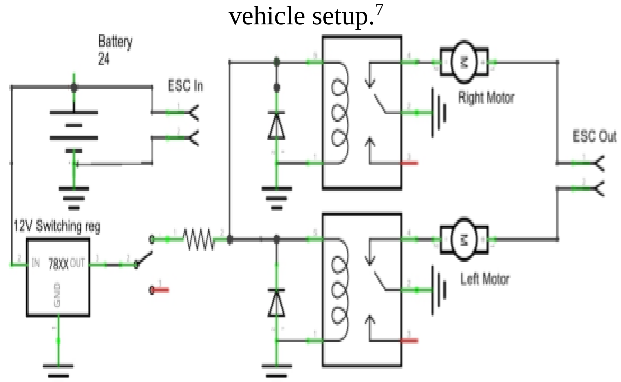


Figure 31. 2016 Louisiana State University e-stop wiring diagram.[18]

Section 2.4. Technical Challenge #4 – Vehicle Machine Vision – Sensor Selection/Processing/Implementation

Vehicle machine vision is a huge part of a successful IGVC vehicle, as the vehicle is completely on its own while operating in the Auto-Nav Challenge. As mentioned above, teams normally use mono/stereo cameras and LADAR. Component redundancy is important, even more-so with regards to sensors, with some teams adding multiple cameras for redundancy as well as to increase the sensors' field of view for detection.

Teams have also installed planar LADARs on pan-tilt assemblies to allow for 3-D sweeping detection. 3-D sweeping is especially important for detecting negative obstacles, like potholes.

Figure 32. 2017 Hosei University vehicle sensor layout.[13]

A significant sensor challenge is not just processing and analyzing a sensor's data feed, but then integrating it with the other vehicle sensors to build a coherent world map of the vehicle's environment. Normally simultaneous localization and mapping (SLAM) algorithms are used for this purpose. SLAM also serves as a good redundancy to the data pulled from the vehicle's high precision differential GPS.

This then immediately ties into requiring robust software coding, building in a comprehensive ruleset to be able to segment out irrelevant data and filter noise, as well as segment and recognize important parts of the world map corresponding to obstacles (barrels, potholes, ramps) and other items of interest (flags, spray painted course boundary lines, etc.). In addition to categorizing these items, there needs to be further logic with regards to flags and spray painted course lines.

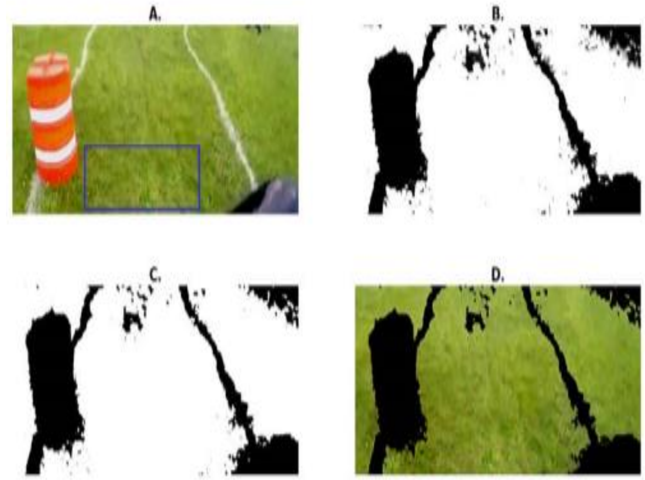


Figure 33. 2016 Stony Brook University vehicle camera extracted histogram projection.[19]

The logic for spray painted lines is straightforward, to have the vehicle stay between the two boundary lines. The logic for flags is more involved, requiring the machine vision system to first not only detect the flags, but accurately determine their color (red or blue), and then after knowing the color, program the vehicle to stay to the left of the red flags and to the right of the blue flags.

Sensor noise can become extremely problematic, requiring implementation of additional processing techniques, such as the 2015 Oakland University team's application of an Artificial Neural Network (ANN) to assist in the determination of the white course boundary lines. Using self-learning approaches can be very helpful in situations like this, where hard coding white line extraction algorithms that will be applicable in real-life IGVC implementation become challenging. The ANN white line detection process the 2015 Oakland University team used is characterized below:

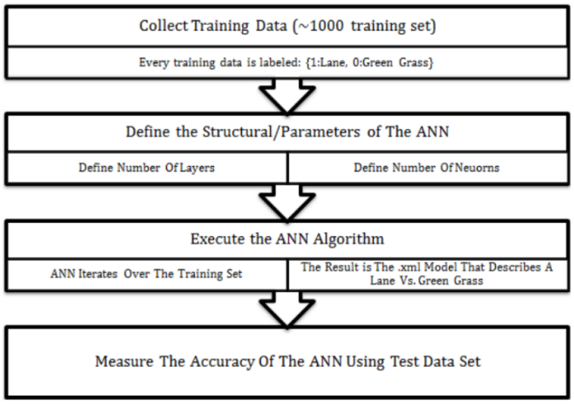


Figure 34. 2015 Oakland University Team's ANN White Line Detection Process[3].

See below for a layout example of the Auto-Nav Challenge course:

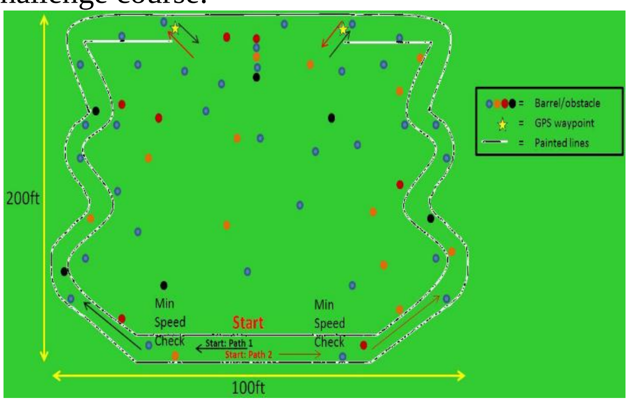


Figure 35. Sketch of Potential 2018 IGVC Auto-Nav Challenge course layout.


Figure 36. Pictures of Auto-Nav Challenge Course.

Section 2.5. Technical Challenge #5 – Vehicle Simulation/Real-Life Testing

Testing of the vehicle is critical and it can take the form of real-life testing and/or simulation. See below for a mock IGVC course created by the 2017 Indian Institute of Technology Bombay team for vehicle testing/evaluation:


Figure 37. 2017 Indian Institute of Technology Bombay mock IGVC course.[8]

An obvious advantage of simulation over real-life testing is that the vehicle can be worked on while evaluating its (virtual) performance on a computer. An obvious drawback to simulations is that it is only as good as the input data, simplifying assumptions, etc. Another advantage of a simulation is that the (virtual) vehicle can be evaluated many times faster than real-time.

The 2015 UNSW team's simulation environment allowed for the simulation to be run up to 5 times faster than real-time and in parallel. The advantages of this can be extreme, assuming wise creation of the simulation environment as a whole and informed determination of the necessary input data, simplifying assumptions, etc., to ensure a highly accurate representation of the real-life vehicle conditions/environment/operations. This can allow for a huge scaling in the amount of vehicle testing within a timeframe, which can greatly improve overall vehicle operation/performance in future real-life testing and at the actual IGVC competition.

Obviously huge amounts of data are generated from these virtual vehicle runs, which then necessitates quick/accurate analysis in order to be useful. For this purpose, the 2015 UNSW team developed and incorporated several tools to "automatically analyze and collect statistics regarding the performance in a simulated run of the competition. These statistics, which include average speed, localization error, and proximity to obstacles, allow for quick tuning and verification of parameters to determine which combination of these parameters optimizes the performance of the system as a whole."[2]

The 2015 CSUN team developed their simulation program using LabVIEW. As they state, "The simulation was developed as a method to allow testing of new codes without endangering the vehicle with a previously untested code, which may have bugs that create unsafe conditions for El Toro...Virtual LRF (laser range finder) data is created, while inducing specified levels of Gaussian white noise to more realistically represent the stream of data that would come from the sensors. This allows the vehicle to choose different paths each time it navigates through the simulation. The simulated data gathered by the LRF and compass is passed to the navigation and system integration code, allowing the vehicle to run autonomously."[1]

The Gazebo simulation environment is especially popular with IGVC teams as can be seen below:

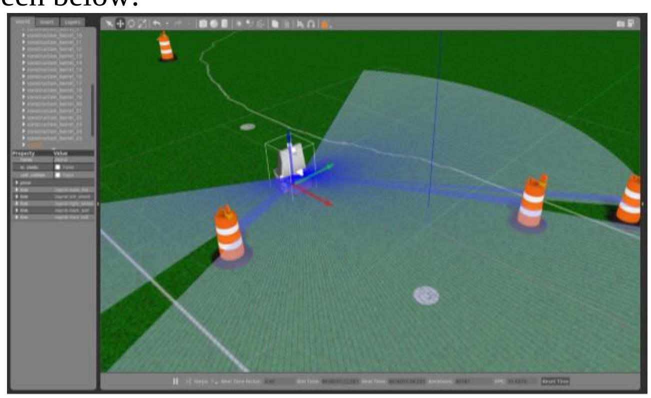


Figure 38. 2017 Georgia Institute of Technology Gazebo simulation.[7]

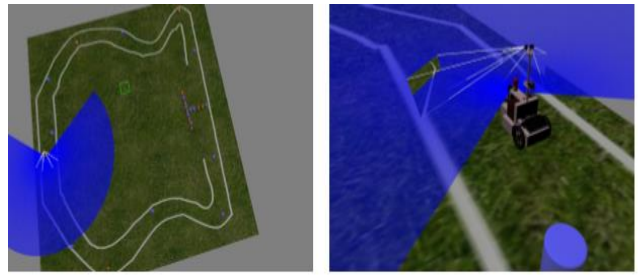


Figure 39. 2017 Indian Institute of Technology Bombay Gazebo simulation.[8]

The real-life improvements of a system, such as for these IGVC vehicles, from utilizing effective simulations cannot be overstated, especially with the growing virtual toolset for improved simulation, analysis and optimization of real-life system performance. Such toolsets include optimization routines such as neural networks and evolutionary systems, as well as deep learning, which was displayed in a limited, though dramatic degree, with regards to a virtual tool (deep learning computer program AlphaGo) quickly optimizing its performance of the game of GO, beating arguably the best GO player in the world, Lee Sedol, well ahead of the projected timeframe, as well as the vast improvements demonstrated by later versions of the deep learning software in successfully reduced time periods (AlphaGo Master/Zero). Deep learning has expanded into many fields including speeding up drug analysis/discovery, self-driving vehicle behavior optimization, etc.

CONCLUSION

The 2017 IGVC was a successful ground vehicle competition which challenged over 30 college teams from around the world to create a functional autonomous vehicle capable of real-world navigation. Teams gained valuable engineering experience which will benefit them in their future careers. The introduction of the Self-Drive Challenge proved a success in providing a highly industry/government relevant competition which further develops the necessary skills engineers should have in the growing fields of autonomy, AI, machine learning, self-driving vehicles, etc.

ACKNOWLEDGEMENTS

[1] Sami Oweis (Ph.D., Embedded Application and Artificial Intelligence Control)

REFERENCES

[1] California State University-Northridge (CSUN) Design Report. 2015. http://www.igvc.org/design/2015/4.pdf.

[2] University of New South Wales (UNSW) Design Report. 2015. http://www.igvc.org/design/2015/24.pdf.

[3] Oakland University Design Report. 2015. http://www.igvc.org/design/2015/16.pdf.

[4] United States Naval Academy (USNA) Design Report. 2014. http://www.igvc.org/design/2014/30.pdf.

[5] IGVC Self-Drive Rules. 2018. http://www.igvc.org/2018selfdriverules.pdf.

[6] Lawrence Technological University Research Day 2018 Presentation. 2018. https://www.ltu.edu/provosts_office/research_day.asp

[7] Georgia Institute of Technology Design Report. 2017. http://www.igvc.org/design/2017/5.pdf.

[8] Indian Institute of Technology Bombay Design Report. 2017. http://www.igvc.org/design/2017/8.pdf.

[9] Oakland University Octagon V3.0 Design Report. 2017. http://www.igvc.org/design/2017/15.pdf.

[10] Roger Williams University Design Report. 2017. http://www.igvc.org/design/2017/17.pdf.

[11] Indian Institute of Technology Madras Design Report. 2017. http://www.igvc.org/design/2017/7.pdf.

[12] Lawrence Technological University Design Report. 2017. http://www.igvc.org/design/2017/11.pdf.

[13] Hosei University Design Report. 2017. http://www.igvc.org/design/2017/6.pdf.

[14] Embry-Riddle Aeronautical University Design Report. 2017. http://www.igvc.org/design/2017/3.pdf.

[15] Michigan Technological University Design Report. 2017. http://www.igvc.org/design/2017/12.pdf.

[16] Oakland University Pinguino Design Report. 2017. http://www.igvc.org/design/2017/13.pdf.

[17] Rochester Institute of Technology Design Report. 2017. http://www.igvc.org/design/2017/16.pdf.

[18] Louisiana State University Design Report. 2016. http://www.igvc.org/design/2016/13.pdf.

[19] Stony Brook University Design Report. 2016. http://www.igvc.org/design/2016/22.pdf.

[20] J. Park and B. Kuipers, "A smooth control law for graceful motion of differential wheeled mobile robots in 2d environment," 2011 IEEE International Conference on Robotics and Automation (ICRA), May 2011, pp. 4896-4902. W.-K. Chen, Linear Networks and Systems (Book style). Belmont, CA: Wadsworth, 1993, pp. 123–135.

[21] Olson, Edwin. "A primer on odometry and motor control." (2004).

[22] J. Park and B. Kuipers, Autonomous Person Pacing and Following with Model Predictive Equilibrium Point Control, 2011.

[23] P. Foster, Z. Sun, J. Park and B. Kuipers, VisAGGE: Visible Angle Grid for Glass Environments *LIDAR mapping for glass, metal, mirrors, and other non-diffuse surfaces, 2011.

[24] K. Iyengar, "Smooth control path planning and tracking of an autonomous mobile robot in real-time," M.S. thesis, School of Eng. And Computer Sc., Oakland Univ., Rochester, Michigan, USA, Aug. 2015. Accessed on: May. 28, 2018.

[25] Hargadine, Calvin S. Mobile robot navigation and obstacle avoidance in unstructured outdoor environments. Diss. Monterey, California: Naval Postgraduate School, 2017.