Literature Review - Investigating the use of Genetic Algorithms to optimise a path planning algorithm within the Context of a Formula Student Team

Alex Dumbreck - 40552841

Contents

0.1 Acronyms and Abbreviations

Acronym	Description
SLAM	Simultaneous Localisation and Mapping
GA	Genetic Algorithm
FSUK	Formula Student UK
AI	Artificial Intelligence
IPO	Initial Project Outline
ROS2	Robot Operating System 2
LiDAR	Light Detection and Ranging
GPS	Global Positioning System
PID	Proportional-Integral-Derivative
EKF	Extended Kalman Filter
UKF	Unscented Kalman Filter
RANSAC	Random Sample Consensus

Chapter 1

Papers and Articles reviewed

1.1 Analysis: Comparison of different SLAM approaches for a driverless race car (Le Large, Bieder, and Lauer, 2021)

1.1.1 Bibliographic Information

- **Title:** Comparison of different SLAM approaches for a driverless race car
- **Authors:** Le Large, Bieder, and Lauer
- **Year:** 2021
- **Type:** Journal/Conference Paper (assumed, based on context)

1.1.2 Abstract/Core Focus

This paper presents a comprehensive comparison of EKF SLAM, FastSLAM, and GraphSLAM for a driverless race car in Formula Student. It evaluates accuracy, computational efficiency, and robustness through simulation and real-world tests on a Formula Student platform. GraphSLAM is found to be most accurate, while EKF SLAM is more resource-efficient. The paper also discusses practical implementation challenges like sensor selection, data association, and dynamic environments.

1.1.3 Methodology

- Systematic comparison using a consistent testing framework (simulation and real-world tracks).
- Modular software architecture for substituting SLAM approaches with consistent sensor inputs (LiDAR, cameras, wheel encoders) and evaluation metrics.
- Sensor calibration and synchronisation emphasised.

- Performance measured against ground truth (high-precision GPS), evaluating map accuracy, localisation error, processing time, and memory across different track layouts/conditions.

1.1.4 Key Findings

- GraphSLAM: Superior accuracy in mapping and localisation (approx. 30% lower position error).
- EKF SLAM: Lowest computational requirements, suitable for resource-constrained systems.
- FastSLAM: Balanced accuracy/performance, but sensitive to parameter tuning.
- All approaches struggled with dynamic obstacles and reflective surfaces.
- Visual feature integration with LiDAR improved localisation in feature-sparse areas.
- Sensor fusion (short-range precision + long-range context) is critical for robustness.

1.1.5 Strengths of the Research

- **Direct Comparison:** Offers a focused comparison of key SLAM algorithms relevant to autonomous racing.
- **Formula Student Context:** Directly applicable to the honours project domain, using a Formula Student platform and addressing relevant challenges.
- **Dual Evaluation:** Combines simulation and real-world testing for comprehensive validation.
- **Practical Insights:** Addresses real-world implementation issues (sensor fusion, data association).
- **Quantitative Metrics:** Provides clear performance metrics for different SLAM approaches.

1.1.6 Weaknesses/Limitations of the Research

- **Software Version Dependency:** Findings tied to specific SLAM library versions and software states in 2021.
- **Hardware Specificity:** Results may be influenced by the specific sensor suite and computational hardware of the KA-RaceIng car.
- **Limited Scope of GAs:** The paper focuses on SLAM, not on the subsequent path planning or GA optimisation, which is the core of the honours project.

1.1.7 Relevance to Honours Project: "Investigating the use of Genetic Algorithms to optimise a path planning algorithm within the Context of a Formula Student Team"

Input Relevance

- **SLAM Algorithm Selection:** Provides critical data for selecting or developing a SLAM algorithm that will produce the map input for the GA path planner. The choice of SLAM (e.g., GraphSLAM for accuracy vs EKF for efficiency) directly impacts the quality and characteristics of the map data the GA will use.
- **Map Characteristics Understanding:** The paper details the expected accuracy, consistency, and potential failure modes of maps generated by different SLAM approaches in a Formula Student context. This informs the GA design regarding how to handle map imperfections or uncertainties.
- **Sensor Data Insights:** Discussion on sensor fusion and data association helps in understanding the raw inputs to the SLAM system, which indirectly affects the final map quality for the GA.

Process Relevance

- **Informing SLAM Component of Project:** If the honours project involves implementing or choosing a SLAM system, this paper is a primary reference for making informed decisions based on performance trade-offs.
- **Understanding SLAM Output for GA:** The GA path planner must be designed to work with the specific type of map output by the chosen SLAM system (e.g., point clouds, feature lists, occupancy grids). This paper helps anticipate what that output will look like.
- **Simulation Environment Design:** The methodologies for testing SLAM in simulation (and real-world) can inform how the simulation environment for the GA path planner should be designed, particularly in how it models sensor data and track features relevant to SLAM.
- **Benchmarking SLAM Performance:** While the project focuses on GA path planning, the performance of the underlying SLAM system is crucial. This paper provides benchmarks for what constitutes good SLAM performance in this context.

Output Relevance

- **Quality of Input to GA:** The success of the GA-optimised path planner heavily depends on the quality of the SLAM-generated map. This paper helps set expectations for that map quality and how it might affect the final path planning performance.
- **System Integration Context:** The paper underscores that SLAM is part of a larger autonomous system. The GA path planner is another

component, and understanding SLAM helps in designing the interface between these components.

- **Justification for SLAM Choice (if applicable):** If a specific SLAM approach is chosen as part of the project, this paper provides strong justification for that choice based on empirical evidence in a relevant context.

1.1.8 Further Notes/Actionable Insights

- Provides valuable insights for SLAM in high-speed autonomous vehicles in structured environments.
- Comparison methodology can be adapted for evaluating other SLAM approaches.
- Findings on computational efficiency are relevant for embedded systems.
- Challenges with dynamic environments highlight considerations for robust localisation.
- Sensor fusion approaches offer practical guidelines for hardware/integration.
- Builds on probabilistic SLAM foundations (Bayesian filtering, EKF, GraphSLAM as least squares, FastSLAM as particle filter with Rao-Blackwellisation).
- Discusses JCBB for data association and the Markov assumption.
- Situated in FSD competition (KA-RaceIng car), noting standardised cone tracks and FSD-specific SLAM requirements (real-time, robustness, high-speed, map/localise phases).
- References core SLAM literature, AMZ Racing publications, and sensor fusion techniques.
- Investigate current versions/advancements in GraphSLAM and EKF SLAM since 2021.
- Consider the trade-off between SLAM accuracy (e.g., GraphSLAM) and the computational budget available for both SLAM and the GA path planner on the target hardware.

1.2 Analysis: Formula Student UK AI Rules

1.2.1 Document Overview

- **Title:** Formula Student UK AI Rules 2025 (relevant sections pertaining to Dynamic Events and ADS requirements)
- **Source:** Formula Student / IMechE (assumed)
- **Year:** 2025 (effective year)
- **Type:** Competition Rulebook

1.2.2 Core Focus/Summary

The Formula Student AI (FS-AI) 2025 rules outline dynamic events (Skidpad, Acceleration, Autocross, Trackdrive) requiring an Automated Driving System (ADS) to navigate cone-demarcated courses. A core ADS requirement is the Dynamic Driving Task (DDT), encompassing Object and Event Detection and Response (OEDR), path planning, vehicle control, and mission tracking. This analysis focuses on the implications for SLAM-based cone mapping and GA-based path planning. Penalties for deviations underscore the need for precision.

1.2.3 Key Rules and Implications for SLAM and Path Planning

For SLAM Algorithm Design

- **Cone-based Mapping:** SLAM must use cones as primary landmarks, including reliable detection, classification, and precise localisation.
- **Real-time Performance:** Continuous map and pose updates are necessary for dynamic racing.
- **Robustness:** Must handle varying lighting/weather (Rule D3).
- **Accuracy and Consistency:** Critical for avoiding penalties (Rule D9) and reliable path planning, especially for multi-lap events.
- **Map Representation:** Output should be easily consumable by the GA path planner (e.g., cone coordinates).
- **Initialization and Loop Closure:** Handle unknown environments and maintain long-term accuracy.

For Genetic Algorithm-based Path Planning

- **Input from SLAM:** Directly uses SLAM-generated cone map for drivable area/constraints.
- **Path Representation:** Suitable path representation for GA evolution (e.g., waypoints, splines).
- **Fitness Function Design:** Crucial for minimizing lap time, adhering to boundaries (avoid D9 penalties), ensuring path smoothness/feasibility, fulfilling event-specifics (lap counts), and safety.
- **Constraint Handling:** Effective management of track limits and vehicle dynamics.
- **Exploration vs Exploitation:** Balance in genetic operators for optimal racing lines.
- **Real-time Adaptation (Advanced):** Potential for path adaptation based on evolving SLAM data.

Specific Dynamic Event Requirements

- **D4 Skidpad:** Precise circular/figure-eight paths; SLAM maps tight geometry; GA fitness check must ensure lap count (D9.1.11) and smooth trajectories.
- **D5 Acceleration:** Simple straight path; SLAM identifies start/end; GA ensures corridor adherence.
- **D6 Autocross/Sprint:** Complex single lap; rapid, accurate SLAM mapping; GA finds optimal line, penalizing errors.
- **D8 Trackdrive (10 laps):** Emphasises SLAM consistency and GA's repeated efficient path planning; potential for map refinement and path adaptation.

1.2.4 Challenges and Considerations

- **Sensor Data Interpretation:** Reliable cone detection/differentiation under varying conditions (D3 Weather Conditions) is foundational for SLAM .
- **Computational Resources:** SLAM and GAs must be optimised for on-board processing.
- **Integration of SLAM and Path Planning:** Tight, low-latency integration is vital.
- **State Estimation Uncertainty:** SLAM uncertainty estimates could inform robust GA decisions (e.g., safety margins).
- **Dynamic Obstacles (Not Explicitly Mentioned for Cones):** Unexpected obstacles would require extensions beyond basic cone navigation.

1.2.5 Relevance to Honours Project: "Investigating the use of Genetic Algorithms to optimise a path planning algorithm within the Context of a Formula Student Team"

Input Relevance

- **Problem Definition:** The rules directly define the operational domain, constraints (cone-based tracks, specific manoeuvres), and objectives (lap times, penalty avoidance) for the path planning algorithm.
- **Performance Metrics:** The rules establish the criteria (e.g., speed, accuracy, completion of laps) against which the GA-optimised path planner will be evaluated.
- **Sensor Data Context:** Understanding the environment (cones as landmarks) informs the type of data the SLAM system (input to the path planner) must provide.

Process Relevance

- **Algorithm Design:** The GA's fitness function must be designed to directly address rule compliance (e.g., minimizing penalties for cone hits, maximizing lap completion).
- **Simulation Environment Design:** The simulation environment created for testing must accurately reflect the track layouts and rules described (e.g., cone placement, track dimensions for Skidpad, Autocross).
- **Testing Scenarios:** The dynamic events (Skidpad, Autocross, Trackdrive) provide specific scenarios for testing the robustness and performance of the integrated SLAM and GA path planning system.

Output Relevance

- **Validation of Path Planner:** The success of the GA-optimised path planner will be measured by its ability to generate paths that allow the ADS to perform successfully in rule-compliant simulations of the dynamic events.
- **Demonstration of Feasibility:** Adherence to these rules demonstrates the practical applicability of the GA approach to the Formula Student AI challenge.
- **Project Deliverables:** The rules shape the requirements for the final path planning solution, ensuring it is tailored to the specific context of a Formula Student team.

1.2.6 Further Notes/Actionable Insights

- The FS-AI rules provide a structured environment for developing and testing autonomous racing algorithms, pushing innovation in real-time perception, mapping, and planning.
- Addressing these rules contributes to broader research in robust SLAM for high-speed navigation and optimal path planning in complex, constrained environments.
- The specific focus on cone-based navigation is relevant to other applications where simple, repeated landmarks define operational areas.
- Ensure the simulation environment accurately models penalty zones and scoring according to D9.

1.3 Analysis: GA F1 racing line review

1.3.1 Bibliographic Information

- **Title:** Evolutionary Design Optimisation for a Formula One Car and Track
- **Author:** Harper

- **Year:** 2024
- **Type:** Journal/Conference Paper (assumed)

1.3.2 Abstract/Core Focus

Harper (2024) explores the application of evolutionary algorithms (EAs) to the complex, dual problem of optimising Formula One (F1) car design parameters in conjunction with track-specific performance strategies. The core premise is the co-optimisation of vehicle parameters (aerodynamics, suspension, etc.) and their interaction with specific racetrack characteristics to minimise lap times, likely through a simulation-based approach where configurations and driving strategies are iteratively evolved.

1.3.3 Methodology (Inferred)

- **Evolutionary Algorithms:** Likely a form of Genetic Algorithm (GA) or similar, involving representation for car design and possibly racing lines/strategies.
- **Simulation Environment:** A sophisticated F1 vehicle dynamics and track simulation for fitness evaluation (lap time).
- **Fitness Function:** Centred on lap time, possibly with other factors like stability.
- **Optimisation Parameters:** F1 car design variables and parameters defining track interaction (racing lines, braking points).

1.3.4 Key Findings (Anticipated)

- Demonstration of EA efficacy in the complex F1 car design and track optimisation search space.
- Identification of non-obvious design trade-offs and synergies for improved lap times.
- Insights into the sensitivity of optimal car setup to specific track features.
- A framework for co-optimising vehicle design and on-track strategy.

1.3.5 Strengths of the Research

- **Direct Relevance of EA Application:** Applies EAs to racing performance optimisation, aligning with the honours project's core technique.
- **Lap Time Optimisation Focus:** Shared goal of minimizing lap time makes the general approach relevant.
- **Complex System Optimisation:** Demonstrates EA capabilities in complex, multi-variable systems.
- **Simulation-Based Approach:** Reliance on simulation for fitness evaluation is a common and relevant practice.

1.3.6 Weaknesses/Limitations of the Research

- **Contextual Differences (F1 vs FS):** Vast differences in vehicle dynamics, resources, and engineering scales limit direct applicability of F1 solutions to Formula Student.

- **Focus on Car Design vs Path Planning:** Primary focus is likely on car physical design/setup, not optimising path planning algorithm output for a given car and SLAM-derived map.

- **Absence of SLAM Integration:** Unlikely to address SLAM for track mapping or uncertainties of cone-based tracks, key to the honours project.

- **Complexity and Resource Intensity:** F1 optimisation and simulations are likely far more complex and resource-intensive than feasible for Formula Student.

1.3.7 Relevance to Honours Project: "Investigating the use of Genetic Algorithms to optimise a path planning algorithm within the Context of a Formula Student Team"

Input Relevance

- **Conceptual Framework:** Provides a high-level conceptual framework for applying evolutionary computation to motorsport optimisation problems, even if the specific inputs (F1 car parameters vs path planning parameters) differ.

- **Problem Structuring:** Illustrates how a complex racing problem can be broken down into parameters suitable for EA-based optimisation, offering insights into how path planning parameters might be similarly structured.

Process Relevance

- **Genetic Algorithm Application for Lap Time Reduction:** Serves as a key reference (cited in IPO [1]) for using GAs to optimise an output (a path in the project's case) for minimal lap time. The project aims to "investigate the use of genetic algorithms to optimise a path planning algorithm's output."

- **Methodological Inspiration for GA Design:** Harper's strategies for GA representation, fitness function (lap time), selection, crossover, and mutation can inform the GA design for path optimisation.

- **Simulation for Evaluation:** Reinforces the need for a robust simulation environment for testing GA-driven optimisations, a core project component.

- **Adaptation Challenge as Learning:** Differences highlight the project's challenge: adapting EA methodologies to a novel domain (SLAM-based path planning for FS). The project's contribution is this adaptation.

Output Relevance

- **Validation of EA Approach:** Harper's work (presumed successful) validates the general viability of using EAs for complex performance optimisation in racing, supporting the choice of GAs for the honours project.
- **Benchmarking Concepts:** The concept of iterative improvement and searching for optimal performance via EAs is directly transferable, even if performance metrics are benchmarked against different criteria (e.g., path efficiency and safety vs F1 car design efficacy).

1.3.8 Further Notes/Actionable Insights

- The primary value of Harper (2024) is its demonstration of EAs as a powerful tool for performance optimisation in competitive racing, not its specific F1 solutions.
- The honours project's challenge and contribution will be tailoring EA principles to the unique constraints of optimising a path from a SLAM algorithm in the Formula Student AI context.
- Focus on adapting general EA methodology and simulation-based evaluation rather than specific F1 design parameters.
- Consider how Harper might have handled multi-objective optimisation if factors beyond lap time were critical, as this could be relevant for path safety/smoothness.

1.4 Analysis: Comparative Analysis of ROS-Unity3D and ROS-Gazebo for Mobile Ground Robot Simulation (Platt, 2022)

1.4.1 Bibliographic Information

- **Title:** COMPARATIVE ANALYSIS OF ROS-UNITY3D AND ROS-GAZEBO FOR MOBILE GROUND ROBOT SIMULATION
- **Author:** Jonathan Thomas Platt
- **Year:** 2022
- **Type:** Thesis (Master of Science, The University of Alabama)

1.4.2 Abstract/Core Focus

This thesis investigates and compares a robotics simulation suite based on the Unity3D game engine integrated with ROS (ROS-Unity3D) against the established ROS-Gazebo simulation suite. The comparison focuses on their architecture, the process of environment creation, resource utilisation, and simulation accuracy, particularly for an autonomous mobile ground robot. The core finding is that ROS-Unity3D presents a viable alternative to ROS-Gazebo. It demonstrates better scalability for larger environments, offers superior shadow quality, and is more adept at real-time LiDAR simulation. Conversely, ROS-Gazebo provides a more streamlined interface with ROS, boasts a wider array of existing sensor plugins, and is more efficient in terms of computational resources when simulating smaller environments.

1.4.3 Strengths of the Research

- **Direct Comparison:** Offers a focused and direct comparison between two leading simulation platforms (ROS-Unity3D and ROS-Gazebo), crucial for researchers and developers selecting tools for robotic simulation.

- **Relevance to Mobile Robots:** The analysis is centred on mobile ground robots, which aligns well with projects involving similar systems, such as Formula Student vehicles.

- **Comprehensive Evaluation Criteria:** The study evaluates the simulators across multiple important dimensions: architecture, environment creation, resource usage, simulation speed, localisation error, and mapping accuracy.

- **Addresses Key Challenges:** Highlights the use of simulation to overcome time, cost, and safety challenges in training and testing autonomous navigation systems.

- **Explores Game Engines as Simulators:** Investigates the trend of repurposing game engines like Unity3D for their flexibility, scalability, and superior graphical fidelity compared to some dedicated simulation suites.

- **Specific Performance Insights:** Provides concrete findings, such as ROS-Unity3D's advantages in large environments and LiDAR simulation, and ROS-Gazebo's strengths in ROS integration and resource efficiency for smaller setups.

- **Detailed Architectural Review:** Delves into simulation hierarchy, coordinate systems, time management, physics engines (Unity's PhysX vs Gazebo's multiple options like ODE), model compatibility (URDF, SDF), user interfaces, and ROS connectivity.

1.4.4 Weaknesses/Limitations of the Research

- **Software Version Dependency:** The findings are tied to the specific versions of ROS, Unity3D, and Gazebo used at the time of the research (2022). Given the rapid development in these platforms, some detailed aspects might have changed.

- **Specificity of Test Cases:** The comparison uses a specific Unmanned Ground Vehicle (UGV) (Jackal UGV) and particular test environments (e.g., HRATC, Agriculture, Lunar Surface). The results might not directly extrapolate to all types of robots or scenarios, such as the high-speed dynamics and specific sensor configurations of a Formula Student AI car.

- **Focus on General Comparison:** While comprehensive, the thesis aims for a general comparison rather than providing a prescriptive guide for optimising or selecting a simulator for a highly niche application like Formula Student AI, which may have unique requirements not fully covered (e.g., very specific aerodynamic effects or tire models if needed).

- **Sensor Plugin Availability:** Notes ROS-Gazebo has more existing sensor plugins, which could be a limitation for ROS-Unity3D if custom sensor development is extensive and time-consuming for a project.

1.4.5 Relevance to Honours Project: "Investigating the use of Genetic Algorithms to optimise a path planning algorithm within the Context of a Formula Student Team"

Platt's (2022) thesis is highly relevant to my honours project, especially for the key deliverable of creating "A simulation environment". The insights from this paper directly inform the choices and development process for this simulator, as mapped to my project's Input-Process-Output (IPO) model:

Input Relevance

My project's **input** phase for the simulation environment involves selecting appropriate technologies and understanding their capabilities. Platt's work provides:

- **Technology Evaluation:** A thorough comparative analysis of ROS-Unity3D and ROS-Gazebo, which are primary candidates for a ROS-based simulation environment compatible with ROS2 (as per my project's requirements).

- **Feature Assessment:** Details on critical features such as physics engine performance (important for simulating vehicle dynamics), sensor simulation capabilities (LiDAR, cameras are crucial for SLAM), graphical fidelity (for realistic testing), and resource usage. This helps in weighing trade-offs, e.g., Unity's graphical fidelity vs. Gazebo's ROS integration maturity.

- **Model Compatibility:** Information on compatibility with standard robot description formats like URDF, which is essential for importing or creating the Formula Student car model.

- **Scalability Considerations:** Insights into how each platform scales with environment size and complexity, relevant for creating varied and realistic test tracks for the Formula Student car.

This allows for an informed decision on the foundational software for the simulation environment, directly impacting the quality and suitability of the inputs to the SLAM, path planning, and genetic algorithms.

Process Relevance

The **process** of my project includes "Research into simulation environments" and "Developing a simulation environment". Platt's thesis contributes significantly by:

- **Foundational Research:** Serving as a key academic reference for understanding the state-of-the-art in ROS-compatible simulators for mobile robots.
- **Design Guidance:** The detailed comparison of architectures, coordinate systems, time synchronisation, and physics engines (e.g., Unity's PhysX vs. Gazebo's options) informs the design choices during the development of my custom simulation environment or the customisation of an existing one.
- **ROS Integration Insights:** Explaining the mechanisms for connecting the simulator to ROS (e.g., ROS for Unity, Gazebo's native plugins) is vital for ensuring seamless data flow for sensor data, control commands, and SLAM/navigation outputs.
- **Benchmarking Ideas:** The methodology Platt uses for comparing resource utilisation, simulation speed, and accuracy (localisation error, mapping accuracy) can inspire the benchmarking and validation processes for my own simulation environment to ensure it meets the project's needs for testing path planning and GA optimisation.

Output Relevance

One of the main **outputs** of my project is "A simulation environment". Platt's work helps ensure this output is robust, well-justified, and fit for purpose:

- **Justification of Choice:** The findings can be used to justify the selection of a particular simulation platform (e.g., choosing Unity for better graphics and LiDAR if those are prioritised, or Gazebo for its ROS ecosystem maturity and resource efficiency in simpler scenarios).
- **Defining Scope and Features:** Understanding the capabilities and limitations of existing tools helps define the scope and essential features of the custom simulation environment tailored for Formula Student AI challenges (e.g., accurate cone detection, track representation).
- **Contribution to Report:** The analysis from Platt's thesis can be referenced in the project report, specifically in the section detailing the development and rationale behind the simulation environment, demonstrating a research-informed approach.
- **Context for Algorithm Testing:** A well-chosen/developed simulator, informed by such comparative studies, provides a reliable platform for testing the SLAM algorithm, the path planning algorithm, and the genetic algorithm's effectiveness in optimising lap times, which are other critical outputs of my project.

In summary, Platt's thesis provides a valuable, in-depth analysis that directly supports the research, development, and justification of the simulation environment component of my honours project, ensuring it aligns with current best practices and technological capabilities in robotic simulation.

1.4.6 Further Notes/Actionable Insights

- Investigate the current state and versions of ROS-Unity3D (particularly ROS TCP Endpoint package or similar) and ROS-Gazebo integration, as development is ongoing in both.

- Specifically evaluate the sensor models discussed by Platt (stereo camera, LiDAR, IMU on Jackal UGV) against the sensor suite planned for the Formula Student AI car in my project.

- Consider the physics engine details (e.g., Unity's PhysX capabilities vs Gazebo's options like Bullet or DART) in the context of simulating potentially high-speed vehicle dynamics for Formula Student.

- If considering Unity, research the effort required for creating or adapting sensor plugins versus using Gazebo's more extensive existing library, factoring in development time.

- Platt mentions RTAB-Map for SLAM; this could be a relevant reference or baseline for the SLAM algorithm development in my project.

1.5 Analysis: AMZ driverless paper

1.5.1 Document Overview

- **Title:** AMZ driverless: The full autonomous racing system

- **Source:** Formula Student / IMechE (assumed)

- **Year:** 2025 (effective year)

- **Type:** Competition Rulebook

1.5.2 Core Focus/Summary

The Formula Student AI (FS-AI) 2025 rules outline dynamic events (Skidpad, Acceleration, Autocross, Trackdrive) requiring an Automated Driving System (ADS) to navigate cone-demarcated courses. A core ADS requirement is the Dynamic Driving Task (DDT), encompassing Object and Event Detection and Response (OEDR), path planning, vehicle control, and mission tracking. This analysis focuses on the implications for SLAM-based cone mapping and GA-based path planning. Penalties for deviations underscore the need for precision.

1.5.3 Key Rules and Implications for SLAM and Path Planning

For SLAM Algorithm Design

- **Cone-based Mapping:** SLAM must use cones as primary landmarks, including reliable detection, classification, and precise localisation.
- **Real-time Performance:** Continuous map and pose updates are necessary for dynamic racing.
- **Robustness:** Must handle varying lighting/weather (Rule D3).
- **Accuracy and Consistency:** Critical for avoiding penalties (Rule D9) and reliable path planning, especially for multi-lap events.
- **Map Representation:** Output should be easily consumable by the GA path planner (e.g., cone coordinates).
- **Initialization and Loop Closure:** Handle unknown environments and maintain long-term accuracy.

For Genetic Algorithm-based Path Planning

- **Input from SLAM:** Directly uses SLAM-generated cone map for drivable area/constraints.
- **Path Representation:** Suitable path representation for GA evolution (e.g., waypoints, splines).
- **Fitness Function Design:** Crucial for minimizing lap time, adhering to boundaries (avoid D9 penalties), ensuring path smoothness/feasibility, fulfilling event-specifics (lap counts), and safety.
- **Constraint Handling:** Effective management of track limits and vehicle dynamics.
- **Exploration vs Exploitation:** Balance in genetic operators for optimal racing lines.
- **Real-time Adaptation (Advanced):** Potential for path adaptation based on evolving SLAM data.

Specific Dynamic Event Requirements

- **D4 Skidpad:** Precise circular/figure-eight paths; SLAM maps tight geometry; GA fitness check must ensure lap count (D9.1.11) and smooth trajectories.
- **D5 Acceleration:** Simple straight path; SLAM identifies start/end; GA ensures corridor adherence.
- **D6 Autocross/Sprint:** Complex single lap; rapid, accurate SLAM mapping; GA finds optimal line, penalizing errors.

- **D8 Trackdrive (10 laps):** Emphasises SLAM consistency and GA's repeated efficient path planning; potential for map refinement and path adaptation.

1.5.4 Challenges and Considerations

- **Sensor Data Interpretation:** Reliable cone detection/differentiation under varying conditions (D3 Weather Conditions) is foundational for SLAM .
- **Computational Resources:** SLAM and GAs must be optimised for on-board processing.
- **Integration of SLAM and Path Planning:** Tight, low-latency integration is vital.
- **State Estimation Uncertainty:** SLAM uncertainty estimates could inform robust GA decisions (e.g., safety margins).
- **Dynamic Obstacles (Not Explicitly Mentioned for Cones):** Unexpected obstacles would require extensions beyond basic cone navigation.

1.5.5 Relevance to Honours Project: "Investigating the use of Genetic Algorithms to optimise a path planning algorithm within the Context of a Formula Student Team"

Input Relevance

- **Problem Definition:** The rules directly define the operational domain, constraints (cone-based tracks, specific manoeuvres), and objectives (lap times, penalty avoidance) for the path planning algorithm.
- **Performance Metrics:** The rules establish the criteria (e.g., speed, accuracy, completion of laps) against which the GA-optimised path planner will be evaluated.
- **Sensor Data Context:** Understanding the environment (cones as landmarks) informs the type of data the SLAM system (input to the path planner) must provide.

Process Relevance

- **Algorithm Design:** The GA's fitness function must be designed to directly address rule compliance (e.g., minimizing penalties for cone hits, maximizing lap completion).
- **Simulation Environment Design:** The simulation environment created for testing must accurately reflect the track layouts and rules described (e.g., cone placement, track dimensions for Skidpad, Autocross).
- **Testing Scenarios:** The dynamic events (Skidpad, Autocross, Trackdrive) provide specific scenarios for testing the robustness and performance of the integrated SLAM and GA path planning system.

Output Relevance

- **Validation of Path Planner:** The success of the GA-optimised path planner will be measured by its ability to generate paths that allow the ADS to perform successfully in rule-compliant simulations of the dynamic events.
- **Demonstration of Feasibility:** Adherence to these rules demonstrates the practical applicability of the GA approach to the Formula Student AI challenge.
- **Project Deliverables:** The rules shape the requirements for the final path planning solution, ensuring it is tailored to the specific context of a Formula Student team.

1.5.6 Further Notes/Actionable Insights

- The FS-AI rules provide a structured environment for developing and testing autonomous racing algorithms, pushing innovation in real-time perception, mapping, and planning.
- Addressing these rules contributes to broader research in robust SLAM for high-speed navigation and optimal path planning in complex, constrained environments.
- The specific focus on cone-based navigation is relevant to other applications where simple, repeated landmarks define operational areas.
- Ensure the simulation environment accurately models penalty zones and scoring according to D9.

Bibliography

[1] Daniel Harper. "Evolutionary Design Optimization for a Formula One Car and Track". PhD thesis. University of Georgia, 2024.

[2] Juraj Kabzan et al. "AMZ driverless: The full autonomous racing system". In: *Journal of Field Robotics* 37.7 (2020), pp. 1267–1294.

[3] Nick Le Large, Frank Bieder, and Martin Lauer. "Comparison of different SLAM approaches for a driverless race car". In: *tm - Technisches Messen* 88.4 (2021), pp. 227–236. DOI: `doi:10.1515/teme-2021-0004`. URL: `https://doi.org/10.1515/teme-2021-0004`.

[4] Jonathan Platt and Kenneth Ricks. "Comparative analysis of ros-unity3d and ros-gazebo for mobile ground robot simulation". In: *Journal of Intelligent & Robotic Systems* 106.4 (2022), p. 80.

[5] Formula Student UK. *Formula Student AI 2025 Rules.* Accessed 13/04/2025. URL: `https://www.imeche.org/docs/default-source/1-oscar/formula-student/2025/rules/fs-ai-2025-rules---v1-0.pdf`.