Global Drone Student Challenge

Rules and Guidelines Document

Contents

1 Challenge Overview

- Global Drone Student Challenge introduces participants to Model-Based Design using Simulink. It offers an opportunity to gain expertise in developing an advanced autonomous line-following algorithm for a quadcopter, utilizing the relevant MATLAB & Simulink toolboxes. Participants can showcase and test their skills on a global stage, as this challenge brings together participants from around the world.
- Participant/teams will work virtually on designing a Minidrone line follower algorithm. Participants will need to use their modeling skills to refine the given Simulink model.
- MathWorks will provide participants with complimentary software to work on the problem statement.

1.1 Pre-work

- It is recommended that participants view all the relevant videos on the challenge webpage, including the MathWorks Minidrone Competition video series to better understand the challenge details.
- Participants are expected to complete MATLAB Onramp, Simulink Onramp, Stateflow Onramp, and Image processing and Computer vision Onramp courses before starting to work on their algorithm.

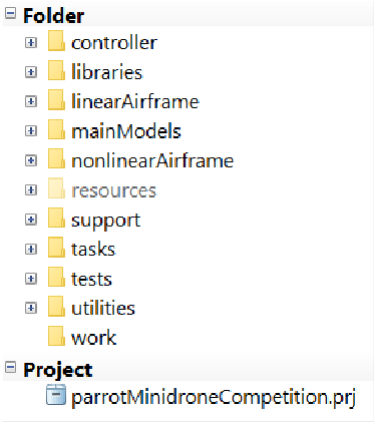
1.2 The Task

- The participant/teams will be provided with a competition model (parrotMinidroneCompetitionStart) and are expected to work on the flight control system block to develop a robust line follower algorithm for the quadcopter using relevant toolboxes available in the MATLAB & Simulink.
- The developed algorithm must be submitted as a project to MathWorks team via the challenge webpage.
- Detailed guidelines for submission and the evaluation process are provided in the following sections.

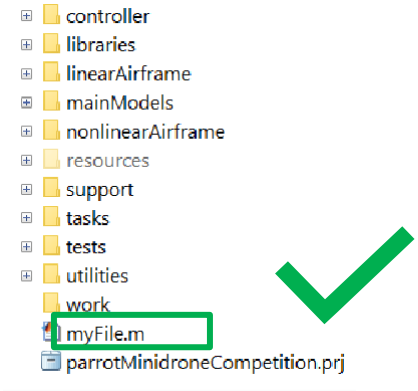
1.3 Submission – Guidelines

- The complete logic for the task completion must be contained in the *parrotMinidroneCompetition* folder that the participant/team submit for evaluations.

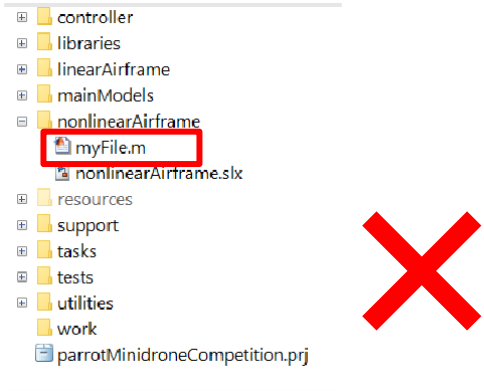
- To submit an entry, the participant/team captain needs to send the Project files to MathWorks. To do so, please use the following guidelines.
  1. **Ensure the correct folder structure:**
     - Ensure that you retain the original file and folder structure from the *parrotMinidroneCompetition* project (from the Simulink Support Package for Parrot Minidrones). The original folder structure looks like the following:



     - Ensure that all the Simulink models and MATLAB files have the same names as in the original *parrotMinidroneCompetition* project.
     - Do not add any Simulink models or MATLAB files to the subfolders. If you have written any additional MATLAB files or Simulink models, you can add them to the main *parrotMinidroneCompetition* model folder. For example, if you have created a new file supporting your model called *myFile.m*, add the file to the folder as shown by the screenshot with the check mark. Do **not** add the files inside a subfolder:

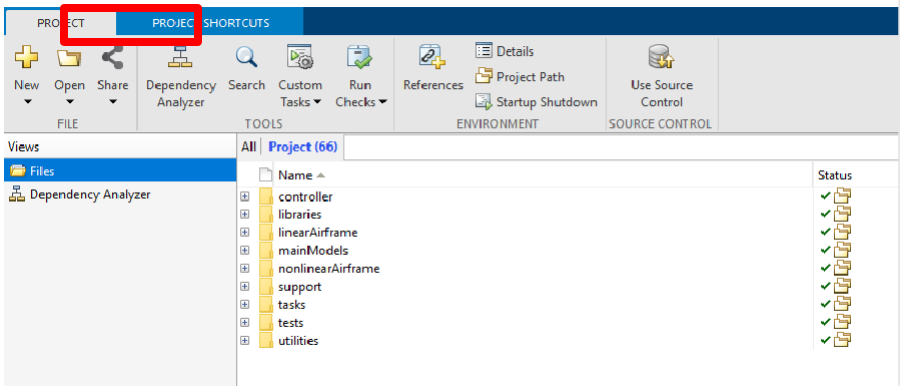


myFile.m outside the subfolders

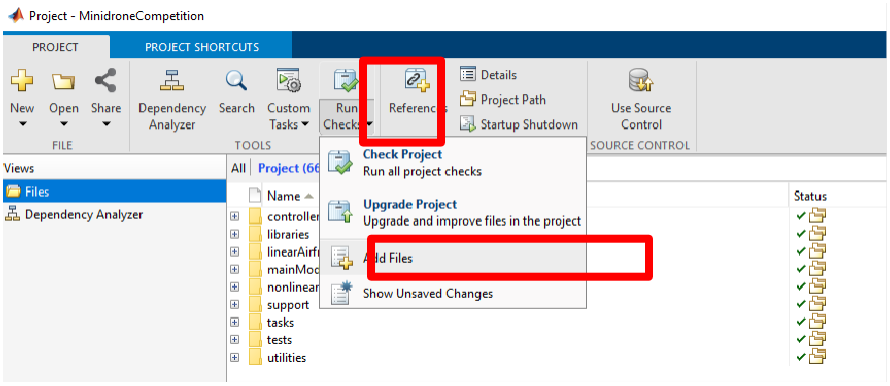


myFile.m present inside a subfolder

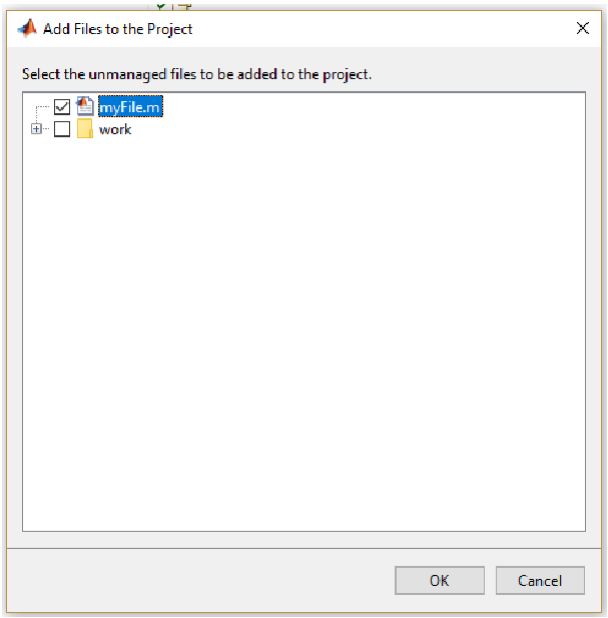
  2. **Add additional files to the project**
     - *Note:* Move to *(3) Submit Project Archive* if there aren't any additional files to be added.
     - Add the necessary files to the Project by navigating to Project tab



- Select Run Checks -> Add Files.

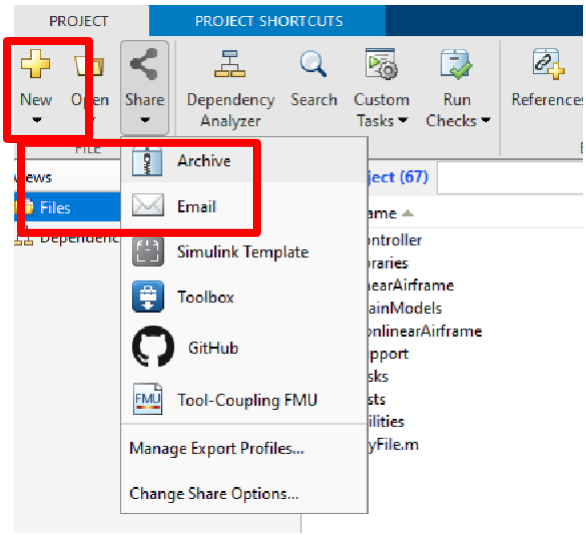


- Check the boxes for the necessary files and click OK.



3. Submit the Project Archive

- In the Projects folder, click on Share and then select *Archive* or *E-mail*.

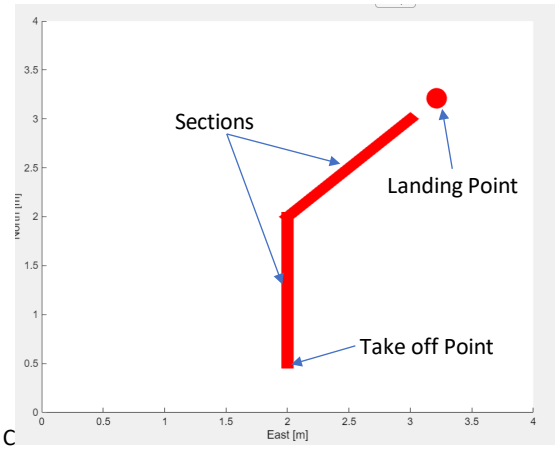


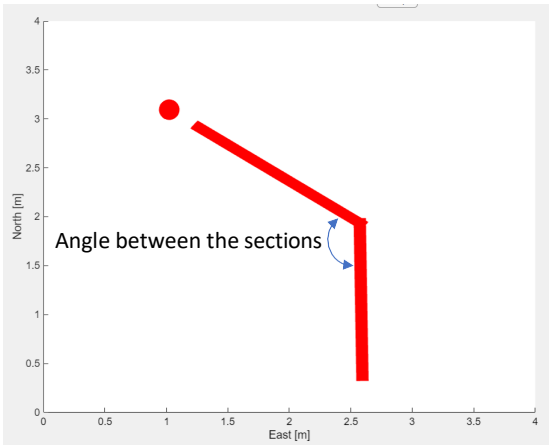
- Select *Export Profile* as 'Whole Project' and name the Project Archive as `<TeamName>` where `<TeamName>` is the name of your team.
- Save the Project Archive (.mlproj) with the naming convention as '`<TeamName>` at `<EventName>` `<EventYear>`' where `<EventName>` is the Global Drone Student Challenge (GDSC) and `<EventYear>` is the year when the challenge launched. For example, if your team's name is 'Drone Squad' and you are participating in the year 2025, your submission would be titled 'Drone Squad - GDSC 2025'.
- The team captain should submit the model using their registered email address. The submission should be made by filling out the 'Submit your entry' form on the challenge webpage.

1.4 Evaluation Process

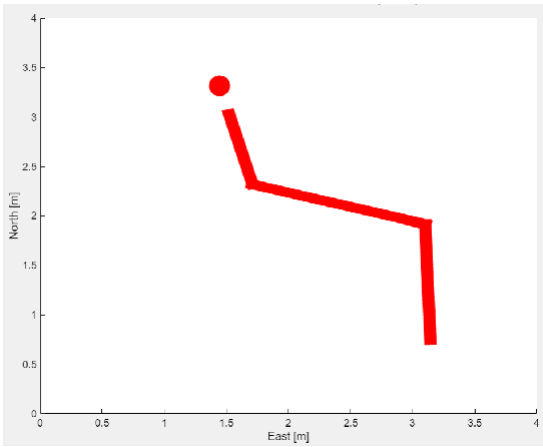
- The submitted model will be evaluated by MathWorks engineers.
- The model will be evaluated based on the capability of completing the line follower track followed by landing on the circular marker.
- The model must be code generation capable.
- The model will be tested across three stages, each with varying levels of track complexity:
  - **Stage 1: Simple Tracks**
  - **Track Structure:**
    - Each track will consist of a minimum of 2 sections and a maximum of 3 sections.
    - The angle between any two connected sections will be obtuse (greater than 90 degrees).

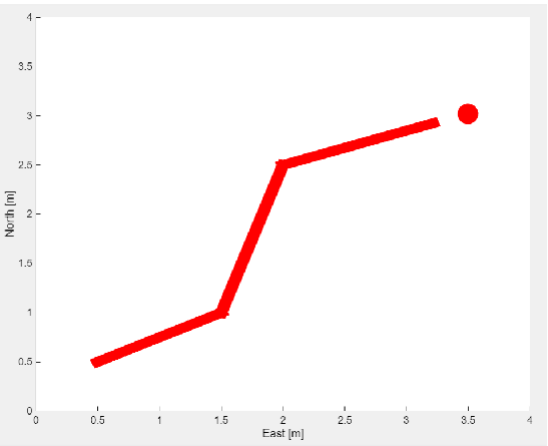
- **Examples:** The following are a few sample tracks for your reference.



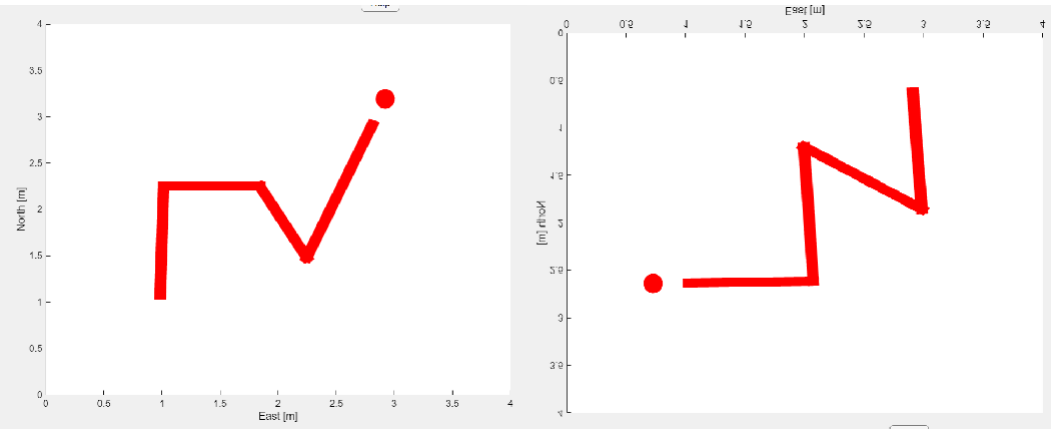


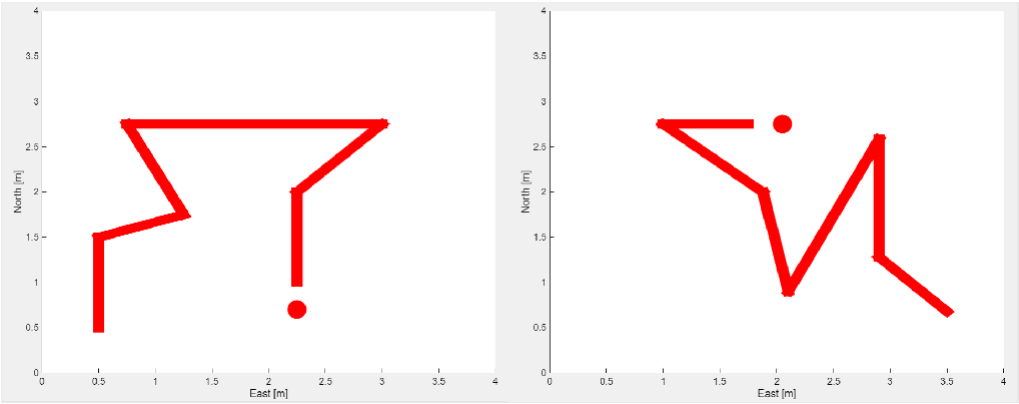
C



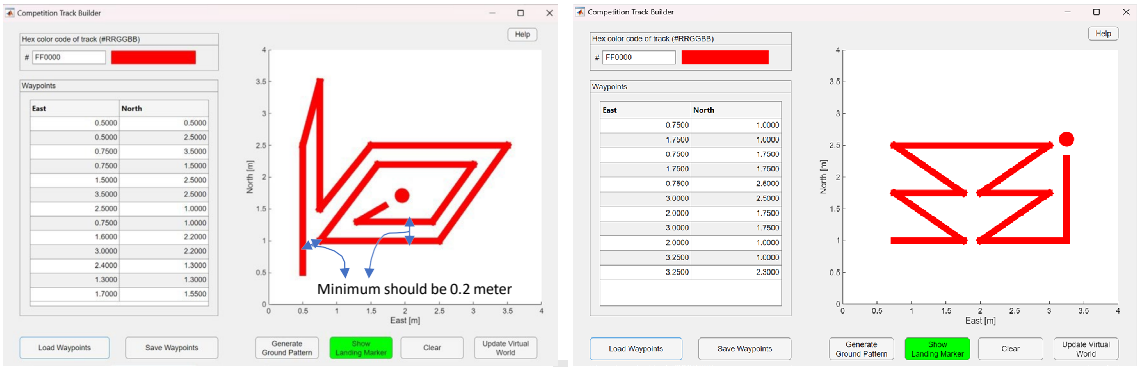


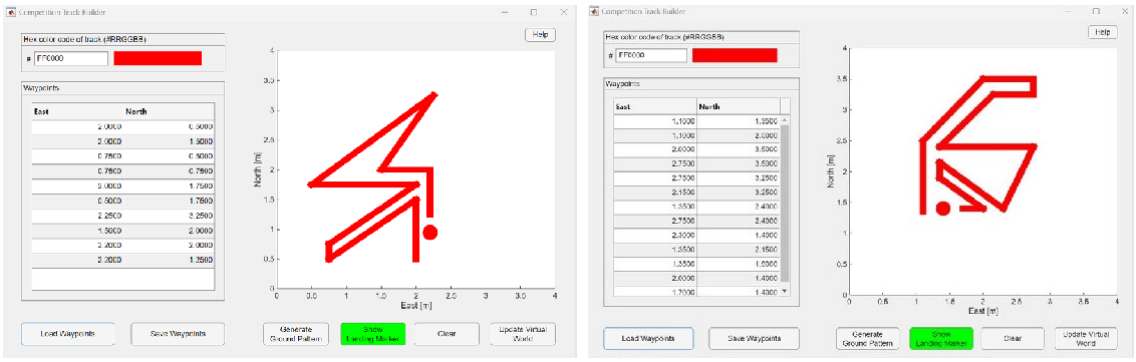
- **Stage 2: Intermediate Tracks**
- **Track Structure:**
    - Each track will consist of a minimum of 4 sections and a maximum of 6 sections.
    - The angle between any two connected sections would range from 15 to 345 degrees.
    - **Examples:** The following are a few sample tracks for your reference





- **Stage 3: Advanced Tracks**
- **Track Structure:**
    - Each track will have a minimum of 7 sections and a maximum of 13 sections.
    - The angle between any two connected sections would range from 15 to 345 degrees.
    - Few of the sections in the track could be in close proximity or in parallel as you can see in the above figures. The minimum distance between any close proximity sections should be 0.2 meters.
    - **Examples:** The following are a few sample tracks for your reference





- Cross sections will not be included in any tracks.
- The color of the track for the simulation round will be Red (#FF0000).
- The model performance will be verified using an internal judging interface prioritizing the algorithm in the following order of importance:
    - Code generation capability of the model
        - Accuracy of the path traced by the drone with respect to the track including soft landing on the circular marker.
        - The number of sections able to complete at different stages of the track
        - Time taken by the drone to complete a track (You can use Simulink Profiler to find the execution time)
- The top three most efficient and accurate algorithms that complete the maximum number of sections in all three stages of tracks in the shortest simulation time will be announced as winners.

2 Rules and Judging

2.1 Rules

- Participants may compete individually or as part of a team of up to four members. If there are any changes in the team structure, the application form for the competition must be resubmitted and an e-mail about the same should be dropped to minidronecompetition@mathworks.com.
- Participants are expected to use either R2024a or R2024b version of MATLAB and related MathWorks products to complete the task. Participants are expected to use R2024a/ R2024b even if R2025a or later versions may be released by that time the submissions are due.

- The decision of the judges will be final.

2.2 Judging

- Scores will be awarded for successfully completing each section of the track, as well as for a soft and accurate landing on the designated circle.
    - For example, if a track has 4 sections worth 5 points each and a soft-landing worth 10 points, the total possible score would be 30 points. The actual points awarded will depend on the accuracy of the line following and soft landing.
- Teams that score 90% or more of the total possible points across Stage 1 and Stage 2 tracks will be eligible to receive a participation certificate.
- The top 3 winners will be selected based on their scores across all the 3 levels of tracks.
- The top 3 winners will be announced via an email and/or social media announcements

3 Reference Material

- Recommended Tutorials:
    - MATLAB Onramp:
        - https://matlabacademy.mathworks.com/
    - Simulink Onramp:
        - https://www.mathworks.com/learn/tutorials/simulink-onramp.html
    - Stateflow Onramp:
        - https://www.mathworks.com/learn/tutorials/stateflow-onramp.html
    - Image Processing Onramp:
        - https://www.mathworks.com/learn/tutorials/image-processing-onramp.html
    - MathWorks Minidrone Competition Video Series
        - https://www.mathworks.com/videos/series/mathworks-minidrone-competition.html
- Additional Video Tutorials:
    - Tech Talk on Drone Simulation and Control:
        - https://www.mathworks.com/videos/series/drone-simulation-and-control.html
    - Programming Drones using Simulink:
        - https://www.mathworks.com/videos/programming-drones-with-simulink-1513024653640.html
    - Tutorials on Computer Vision and Code Generation:
        - https://www.mathworks.com/academia/student-competitions/tutorials-videos.html

    - Tech Talk on State Machines:
        - https://www.mathworks.com/videos/tech-talks/state-machines.html
    - Tutorials on Stateflow:
        - https://www.mathworks.com/videos/series/stateflow-tutorials-94460.html

- Documentation Links:
    - Simulink Support Package for Parrot Minidrones:
        - https://www.mathworks.com/hardware-support/parrot-minidrones.html
        - https://www.mathworks.com/help/supportpkg/parrot/index.html
    - Aerospace Blockset:
        - https://www.mathworks.com/products/aeroblks.html
    - Simulink 3D Animation:
        - https://www.mathworks.com/products/3d-animation.html
    - Stateflow:
        - https://www.mathworks.com/products/stateflow.html
    - Color Thresholder App:
        - https://www.mathworks.com/help/images/ref/colorthresholder-app.html