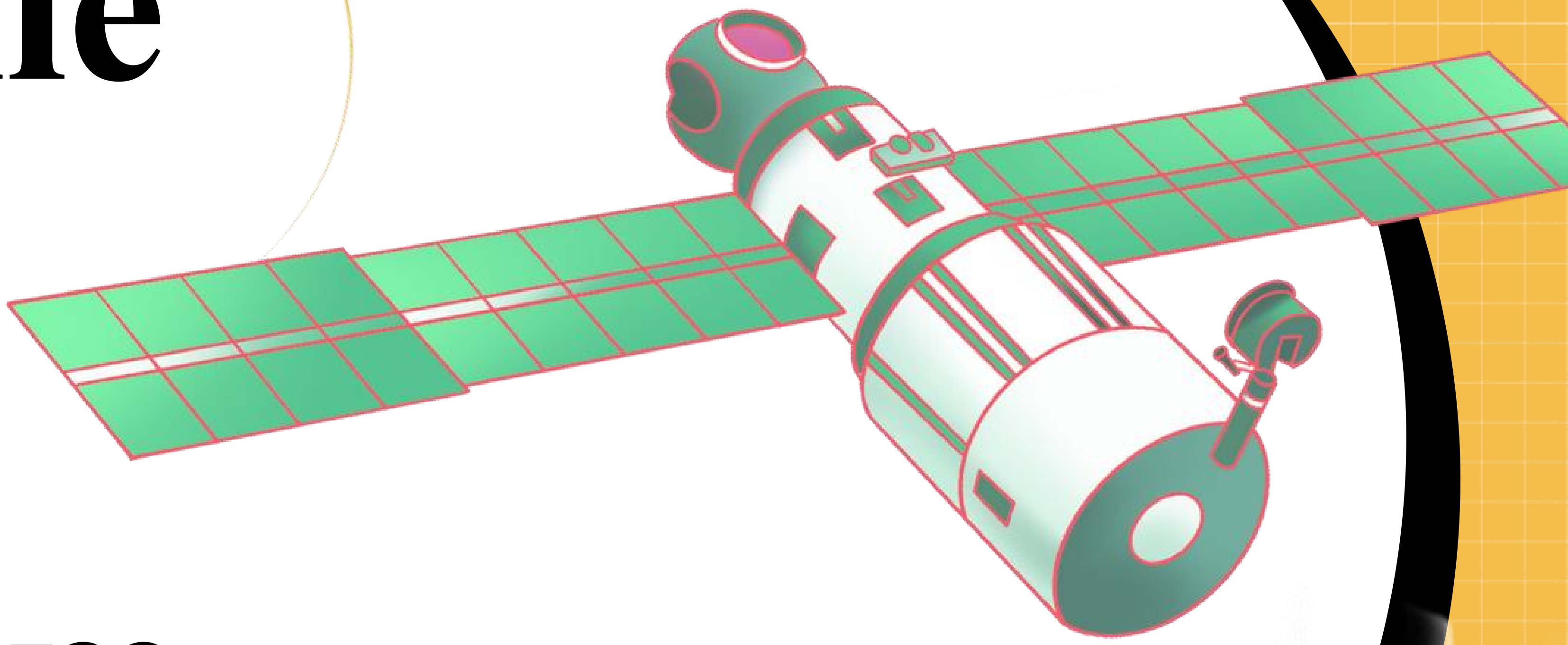
北京郵電大學
Beijing University of Posts and Telecommunications

Autonomous mobile vehicle equipped with 2D lidar system



Team members:

Cao Ziqian, Zhu Hongjia, Wang Zihuo, Long Zijie, Wang Yuxin, Fang Zhengshen, Liu Shuo, Zhang Yiying, Zou Junzi, Liu Jiayi, Wang Shizhe

QMUL

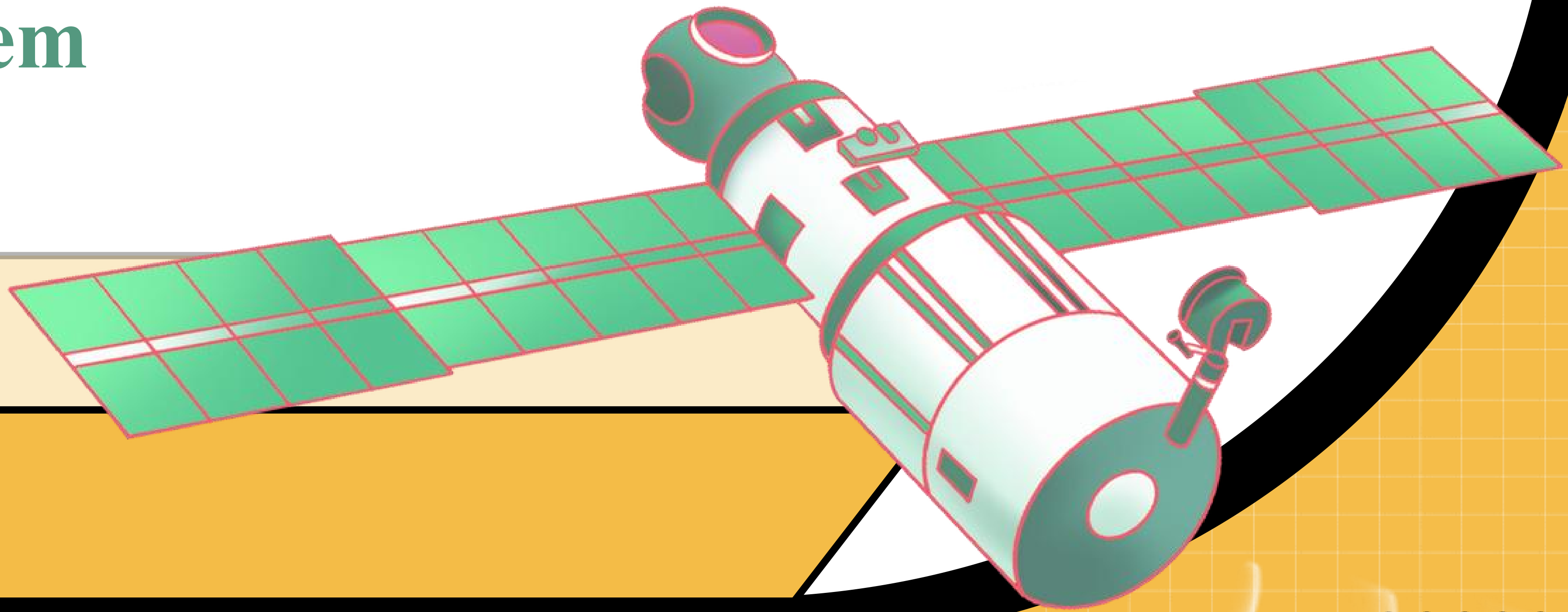
北京郵電大學

Contents

1. Project Division

QMUL

——Autonomous mobile vehicle equipped with 2D lidar system



Cao Ziqian: Communication between Bluetooth Module and System Module

Zhu Hongjia & Wang Zihe: Odometry and Position Estimation

Wang Yuxin: Motor Drive and Control

Long Zijie: Sensor Data Acquisition

Fang Zhengshen: Motion Control and Integration of the Motor Drive Module

Zhang Yiying & Liu Shuo: SLAM Mapping

Zou Junzi: Shortest Path

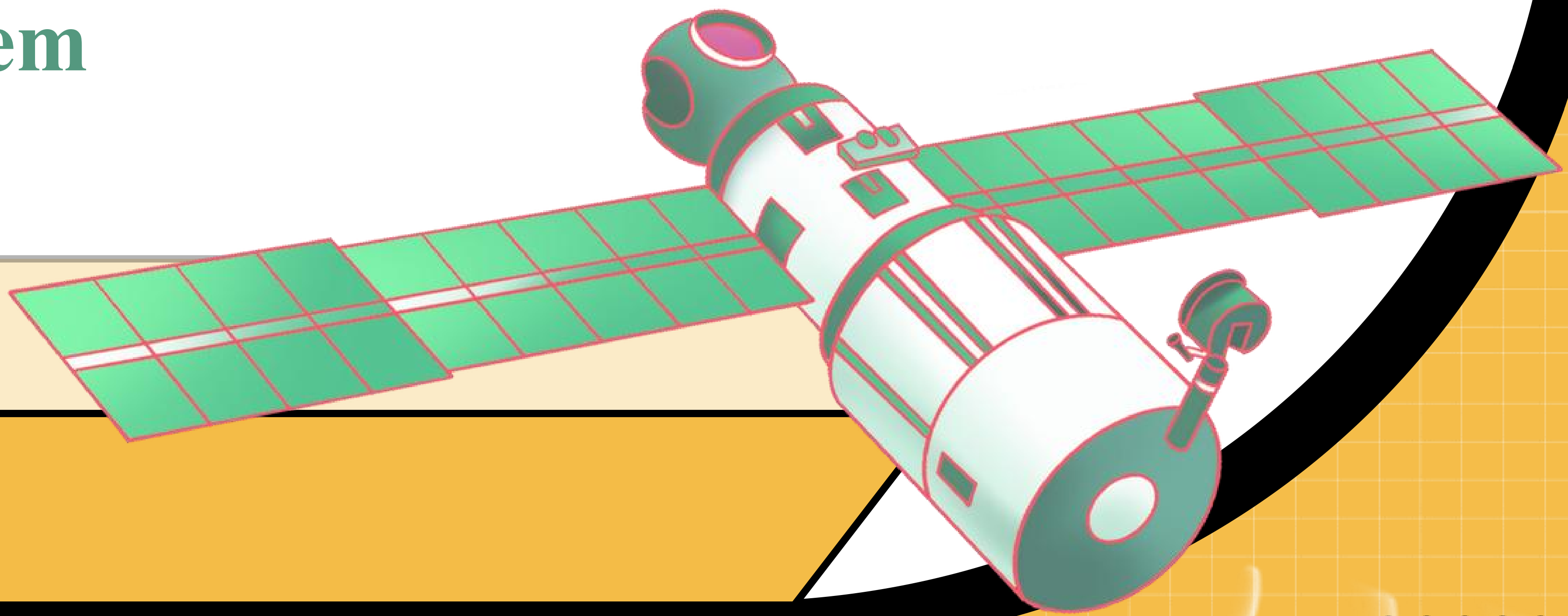
Wang Shizhe: Path Planning

Liu Jiayi: GUI and Visualization

2. Features and Innovations

QMUL

——Autonomous mobile vehicle equipped with 2D lidar system



Sensor data collection

1. Interface with 2D LiDAR sensor

Collect continuous 360-degree environmental sensing data (angles and distances) and send the raw data to the host for SLAM mapping, visualization, path planning, and autonomous navigation.

Sensor data collection

2. Interface with MPU6500 IMU

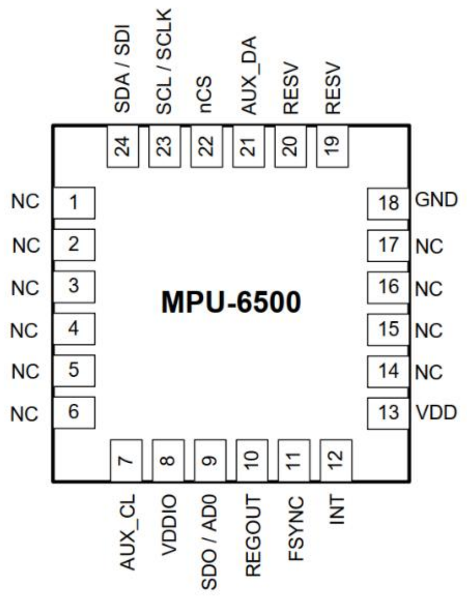
Read directional data, including accelerometer and gyroscope readings.

3. Interface with motor encoder

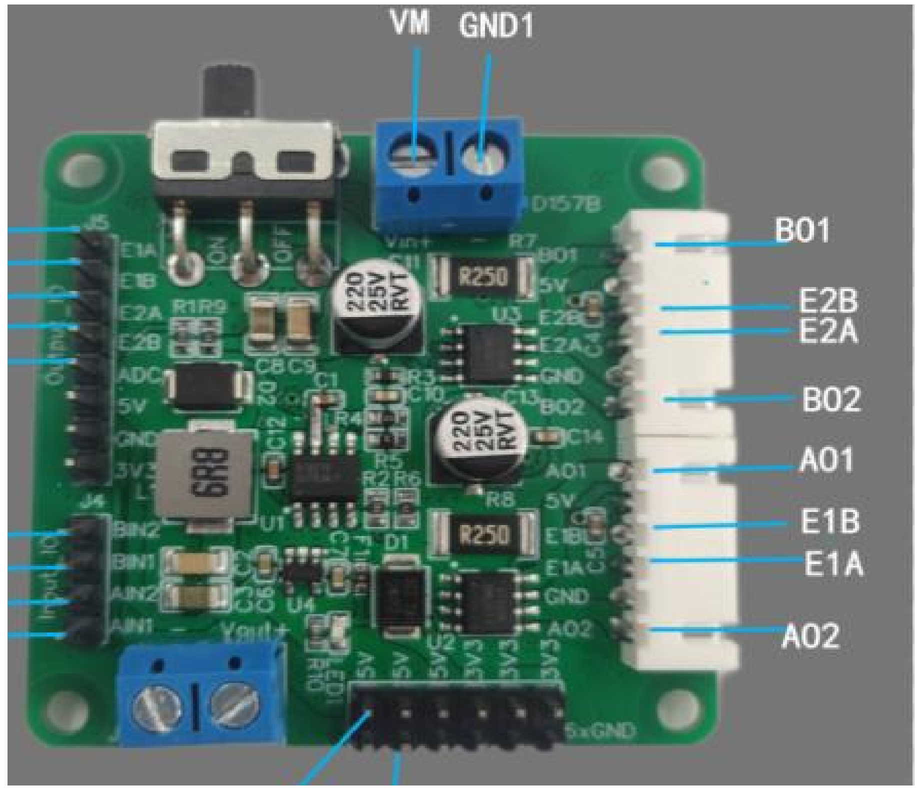
Read motor rotation pulses

Distance and location

The odometer data is calculated using the collected data (IMU and motor encoder), i.e., the robot's pose (x, y, θ), where θ is the robot's heading expressed in radians, for direction/position estimation and correction.

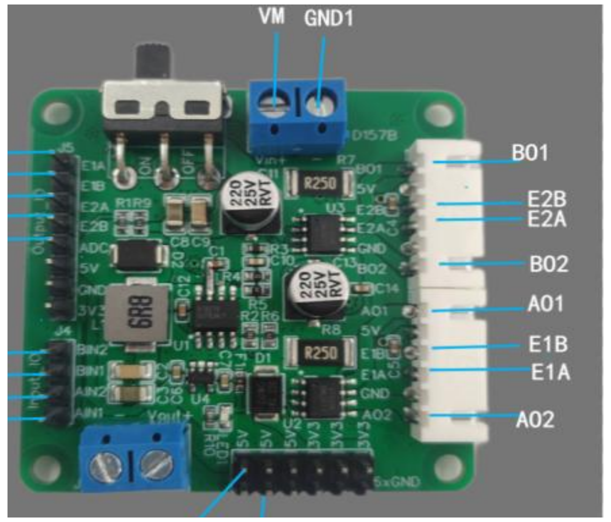


Electric machine control



- Because the output current of the development board is mA level, the current is too small to drive the motor directly, so a DC motor drive module is needed to provide the working current for the motor.
- By adjusting the magnitude of the DC voltage applied to the motor, the rotation speed of the motor can be changed to change the speed of the car; by changing the polarity of the DC voltage applied to the motor, the rotation direction of the motor can be realized.
- The 520 encoder motor employs a Hall encoder to measure the motor's rotational speed and distance traveled, enabling PID closed-loop control.

Electric machine control



520 gear encoder DC motor

· The 520-tooth gear encoder DC motor is controlled by the AT8236 motor driver.

· The two-phase pulse signals from the A/B encoder enable detection of the gear motor's rotation direction and speed. By integrating the PID algorithm, the system calculates the required PWM input to regulate the motor's speed.

AT8236 motor driver

· Configure TIM2 in encoder mode with PA0 connected to channel 1 and PA1 to channel 2;

· Configure TIM4 in encoder mode with PB7 to channel 2 and PB8 to channel 1;

· Use TIM3 for PWM output, with four channels connected to PA8, PA7, PB0, and PB1 respectively;

Bluetooth communication

2D LiDAR sensor

◆ **SLAMTEC RPLIDAR C1：**

The 2D LiDAR sensor, utilizing laser time-of-flight ranging technology, captures angular and distance data of the surrounding environment, enabling real-time mapping and SLAM.

Serial Bluetooth module

◆ **The serial-to-Bluetooth module enables wireless communication between the robot and the host:**

·**Transmit data packets to the host via Bluetooth, including 2D lidar scanning data (angle and distance) and odometer data (x, y, θ);**

·**Receive navigation commands (route points/tracks) from the host to perform real-time motor control.**

SLAM mapping

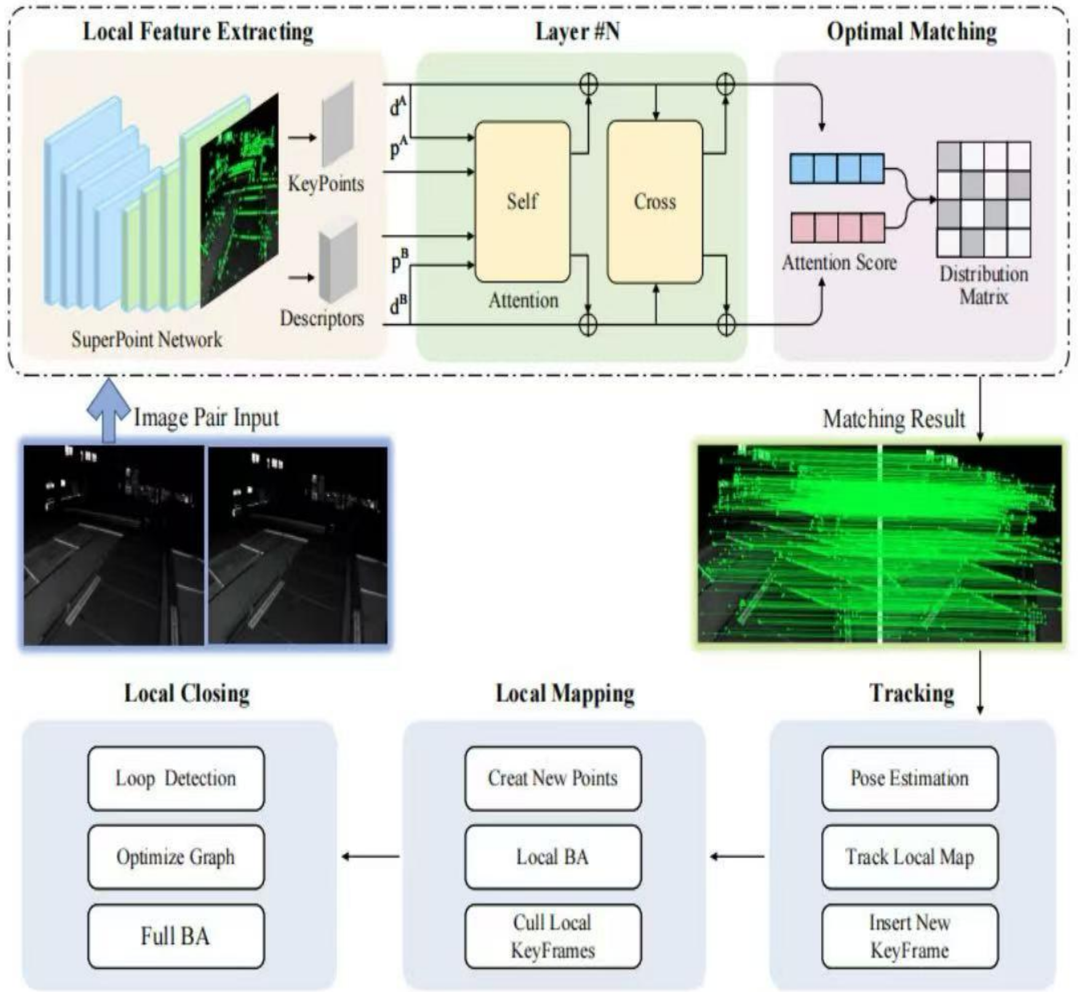
SLAM refers to the technology in which a robot, in an unknown environment, uses its own sensors to acquire environmental data in real-time while simultaneously performing self-localization and environment map construction.

Core Value:

Solving the core problem of 'where the robot goes, where it is, and what the environment is like' in unknown environments is the foundation for autonomous mobile robots to achieve autonomous navigation.

System Feature

Based on a LiDAR odometry hardware solution, it offers integrated capabilities for real-time mapping, autonomous exploration, and path planning, suitable for low to medium-speed indoor and outdoor mobile scenarios.



Visualization Layer and Decision Layer

Visualization Layer

Visualization Layer: The "display window" of the system- Core Functions: Real-time display of the grid map (distinguishing free/obstacle/unknown areas with different colors), the robot's current pose (marking the robot's position and orientation on the map), raw LIDAR scan data (point cloud display), and exploration path planning results (trajectory marking).

Decision Layer

The "command center" of the system- Autonomous Exploration: Based on the frontier algorithm, intelligently identifies the "boundary points between known and unknown areas" on the map, prioritizing frontier points that are closer and cover a larger unknown area as target points, achieving full-scenario mapping without human intervention.

Shortest Path (Core Logic of A* Algorithm)

Node Representation: Each node contains coordinates, actual movement distance from the start node, heuristic distance to the target node, total cost, and a parent node

Search Process:

Maintain an "open list“ and start from the initial node, select the node with the smallest cost from the open list. For each neighbor node, calculate the new cost. If a better path is found, update the node information until reaching the target.

Path Reconstruction: Backtrack from the target node to the start node using parent pointers to form the complete path.

Applications:

Global Path Planning: In a known grid map, it plans collision-free shortest paths based on start and target coordinates.

Safe Path Guarantee: Through the "obstacle expansion" mechanism,obstacles are expanded outward by a certain radius, ensuring the safety of the planned path.

Path Optimization: The smooth_path method uses the algorithm to check if a straight line between two points is obstacle-free, removing redundant intermediate points to generate a smoother path and reduce robot turning frequency.

Innovative Features of the Algorithm

Safe Path Mechanism:

When expanding obstacles, it specifically protects areas around the start and target points. This solves the problem of expanded obstacles.

Diagonal Movement Optimization:

When allowing diagonal movement, it additionally checks if adjacent nodes along the diagonal are obstacle-free and ensures path validity.

Efficient Search Assistance:

Uses the search assistance to quickly locate nodes in the open list, avoiding redundant calculations. Combined with a priority queue,it efficiently retrieves the node with the smallest cost, improving search efficiency.

Path Planning Engine: Finding the Optimal Route

Our robot's ability to navigate the maze intelligently is powered by a robust and multi-layered path planning system. The core of this system is the renowned A (A-Star) search algorithm.

Core Algorithm: A Search

We implemented the A* algorithm to find the shortest possible path between any two points on the grid map generated by the SLAM system.

It intelligently balances the actual distance traveled from the start (g_cost) with an estimated distance to the goal (h_cost - using Euclidean distance), ensuring it finds the most efficient route without searching the entire map.

INNOVATION 1: Safe A Planning with Obstacle Inflation*

Standard A* finds a path but doesn't account for the robot's physical size. A path hugging a wall is optimal but dangerous.

Our Solution: We developed a "Safe A*" planner (a_star_safe). Before planning, it creates a temporary "virtual" map where all obstacles are expanded by a safety radius (100mm).

Benefit: The A* algorithm then plans a path on this inflated map, guaranteeing that the generated route always maintains a safe, collision-free distance from all known walls and obstacles.

A calculated path is just a series of points. Turning it into smooth and safe movement requires further intelligence. We implemented two key features to enhance navigation robustness and efficiency.

INNOVATION 2: Path Smoothing for Efficient Movement

Problem: Grid-based paths from A* are often jagged with unnatural diagonal or "staircase" movements.

Our Solution: After a path is found, a smoothing algorithm post-processes it. It iteratively removes unnecessary intermediate waypoints if a direct, obstacle-free line of sight exists between the start and end points of a segment.

Benefit: This results in shorter, straighter paths, leading to faster travel times and more natural-looking robot motion.

INNOVATION 3: Hybrid Control - Global Path & Local Awareness

We combined our global A* planner with a local, reactive obstacle avoidance system for maximum safety.

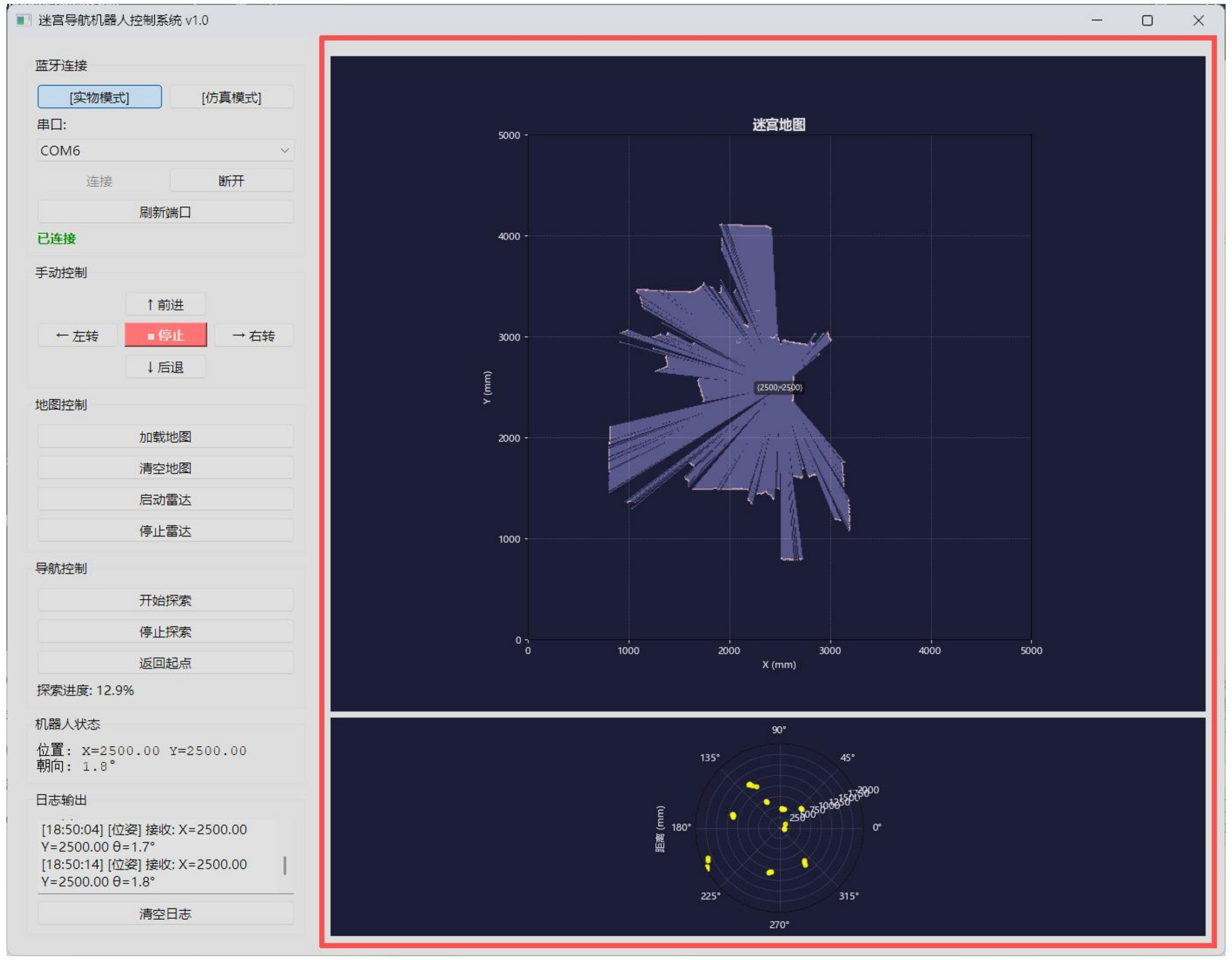
Global Plan (A*): Provides the optimal, long-distance route to the destination.

Local Reaction (check_obstacle_ahead): While following the global path, the robot continuously uses its sensor data to check for immediate obstacles within a 300mm forward-looking cone.

Benefit: This dual approach allows the robot to follow an efficient master plan while being able to instantly stop for unmapped or dynamic obstacles. It is the key to our robot's robust and adaptive behavior in unpredictable environments.

Concise interface design: better user experience

The image below shows a graphical user interface (GUI) titled “Maze Navigation Robot Control System v1.0.” It is divided into two main sections: the left control and information panel and the right real-time visualization area.

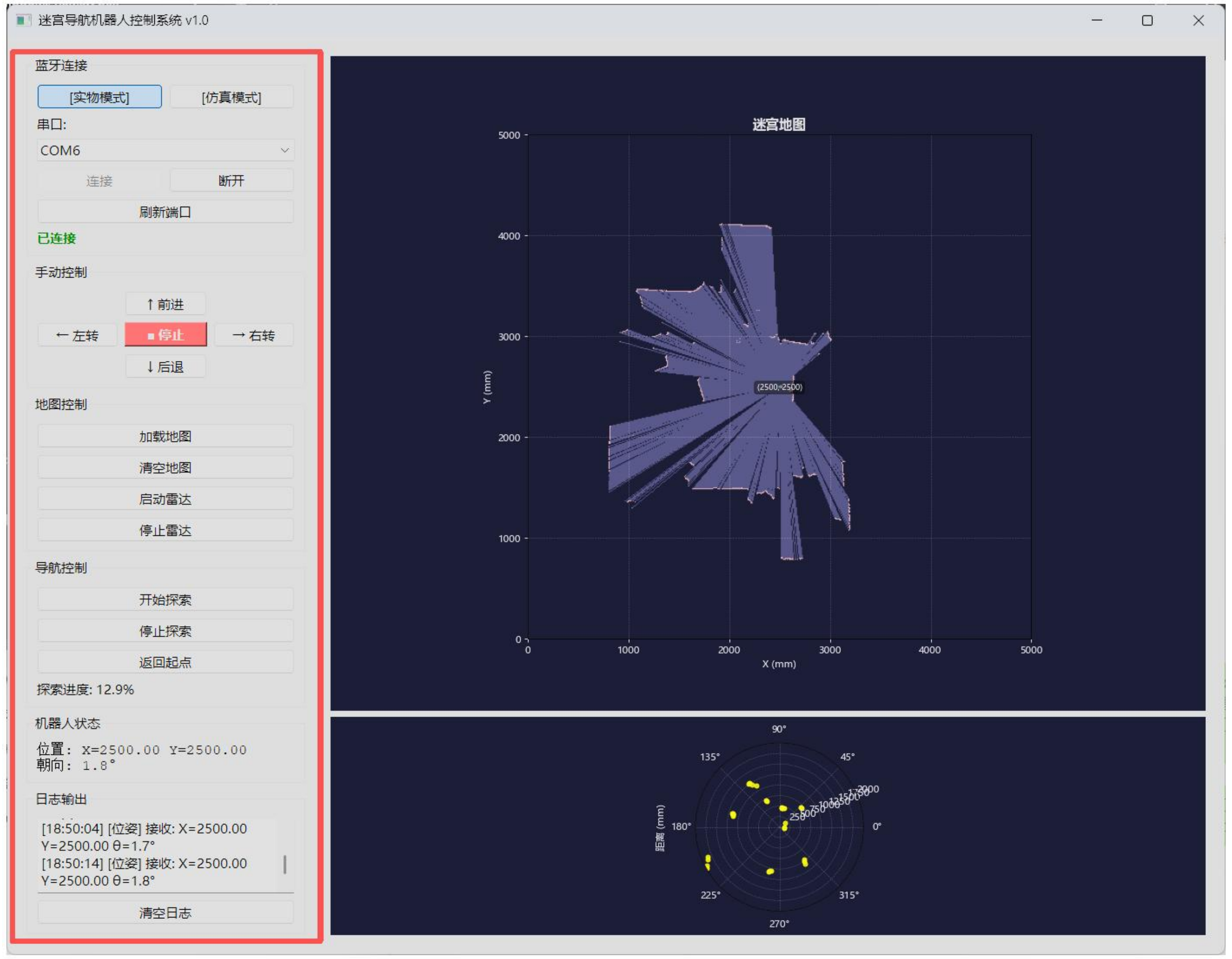


THE RIGHT PART: The right visualization area is divided into two parts. The upper section, titled Maze Map, dynamically generates a 2D environmental map based on Lidar scanning data. The lower section displays a polar coordinate chart, showing the distribution of real-time Lidar distance measurements. Together, these provide an intuitive representation of the robot’s surrounding environment.

INNOVATIONS: The system’s innovations include the integration of dual physical-simulation modes, real-time mapping and navigation control, Bluetooth communication, and an intuitive GUI interface. It not only demonstrates SLAM-based mapping but also supports autonomous exploration and return functions, embodying a complete loop from environmental perception to intelligent decision-making.

Concise interface design: better user experience

At the bottom, the system displays the robot's real-time status, including position coordinates (X, Y), heading angle (θ), and current exploration progress. It also features a log output panel that records timestamps, position data, and angle information for debugging and analysis.



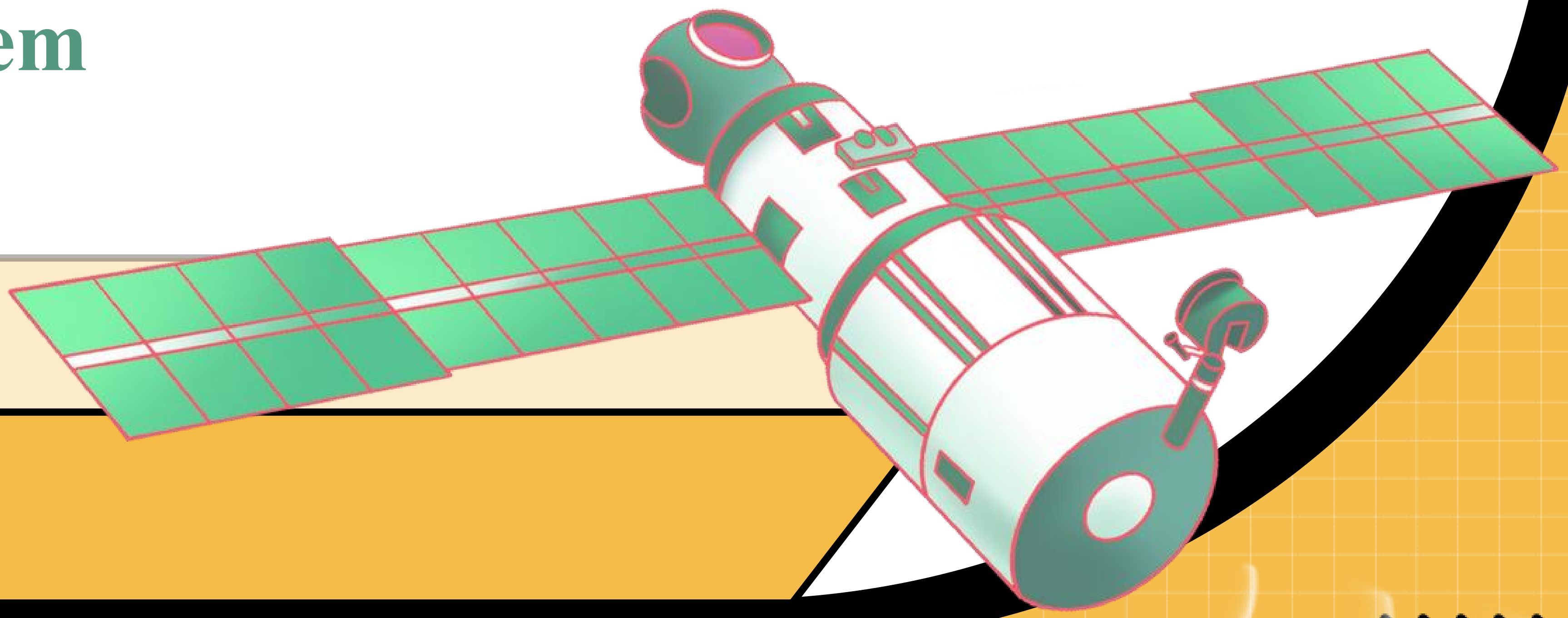
THE LEFT PANEL: This part is mainly used for robot connection, control, and status monitoring. At the top, there are Bluetooth connection settings, allowing users to select between physical mode or simulation mode and choose a communication port (e.g., COM6). The connection status is intuitively displayed through colored text, such as green for "Connected." Below this section, the manual control area includes buttons for forward, backward, left turn, right turn, and stop, enabling direct control of the robot's movements.

The map control section in the middle provides functions such as Load Map, Clear Map, Start Lidar, and Stop Lidar, which are used to obtain and visualize mapping data. The navigation control area supports operations like Start Exploration, Stop Exploration, and Return to Start, enabling autonomous navigation and return-to-origin capabilities.

3. Problems and solutions

QMUL

——Autonomous mobile vehicle equipped with 2D lidar system



Problems and Solutions (Example)

Problems	Solutions
Chassis assembly of the car: It was found that the position and size of the holes on the chassis of the car were not accurate enough. For example, when installing the power drive module, it was found that the position of the holes on the chassis did not match the size of the power drive module.	Use the tool flexibly to get a hole that can be used, and correctly install the power drive module and fix it on the base plate.
Conflicting or misconfigured pin assignments: ① Multiple devices sharing the same pins (e.g., UART/SPI interface conflicts); ② Incorrect pin configurations (e.g., mistakenly connecting PWM pins as GPIO).	Refer to the data sheet to verify pin functions, then create a pin assignment table. In STM32CubeMX, configure pin modes strictly according to this table. Before generating code, check the 'Pinout View' for any conflict warnings.

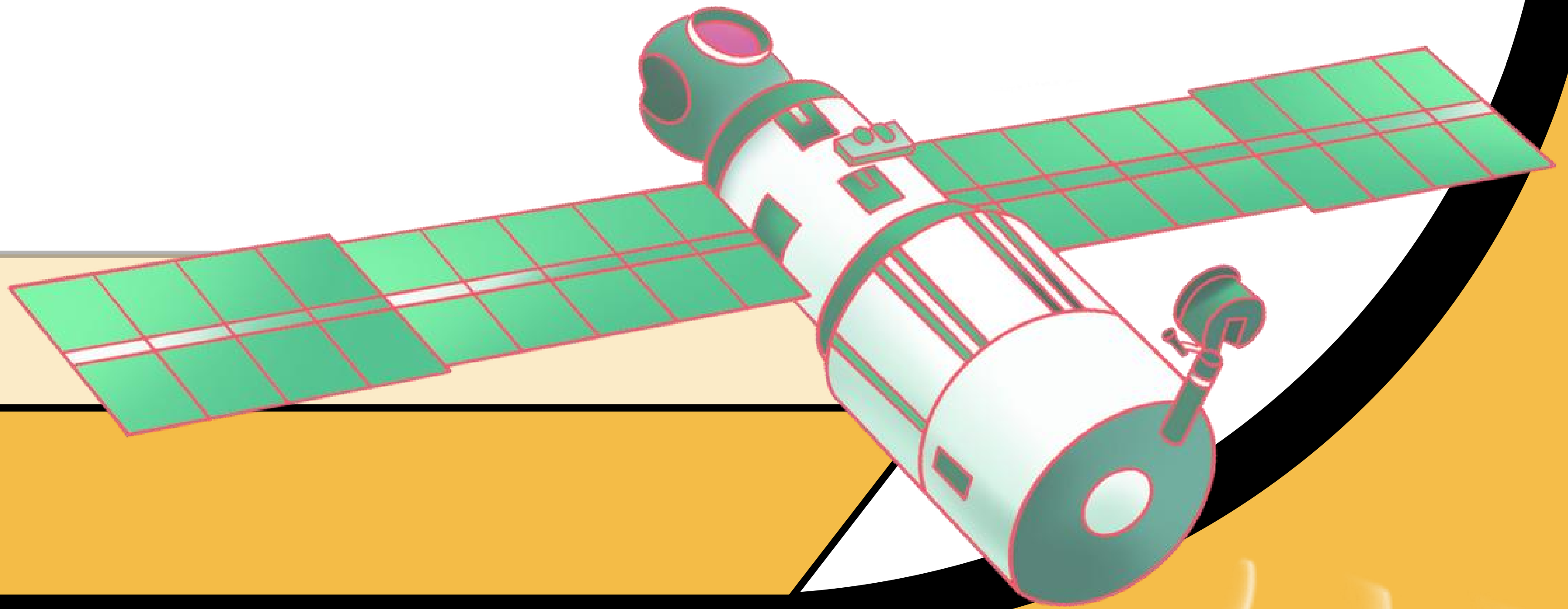
Problems and Solutions (Example)

Problems	Solutions
The Bluetooth module cannot receive data: The data from the cart cannot be sent, and the computer cannot receive it. Reasons: ST-link not downloaded; data format error (not packed according to the protocol: frame header + data + check bit).	Download and update the ST-Link; define a simple communication protocol: frame header 0xAA+ data length + command + parity bit to prevent data packet sticking or misinterpretation.
Code compilation errors (redefinition, undeclared, etc.): The debugger shows 'redefinition of xxx' or 'unknown type name xxx'. Cause: Global variables (e.g., left_pid) are redundantly defined in multiple.c files. Header files are missing (e.g., using UART_HandleTypeDef without usart.h).	· Define global variables in.c files and declare them externally in.h files (e.g., extern PID left_pid); · Ensure the header file contains: main.h peripheral handle definitions, pid.h PID structure definitions; · Enable header protection macros (#ifndef XXX_H) to prevent duplicate inclusion.

QMUL

4. Course Benefits

——Autonomous mobile vehicle equipped with 2D lidar system



Course Benefits

Skill Upgrade

Master STM32 peripheral configuration (CubeMX+Keil)

Skill Upgrade

Understanding hardware system integration methods (power/signal/mechanical)

Skill Upgrade

Significantly improve problem summarization ability

Teamwork experience

Clear division of labor and responsibility

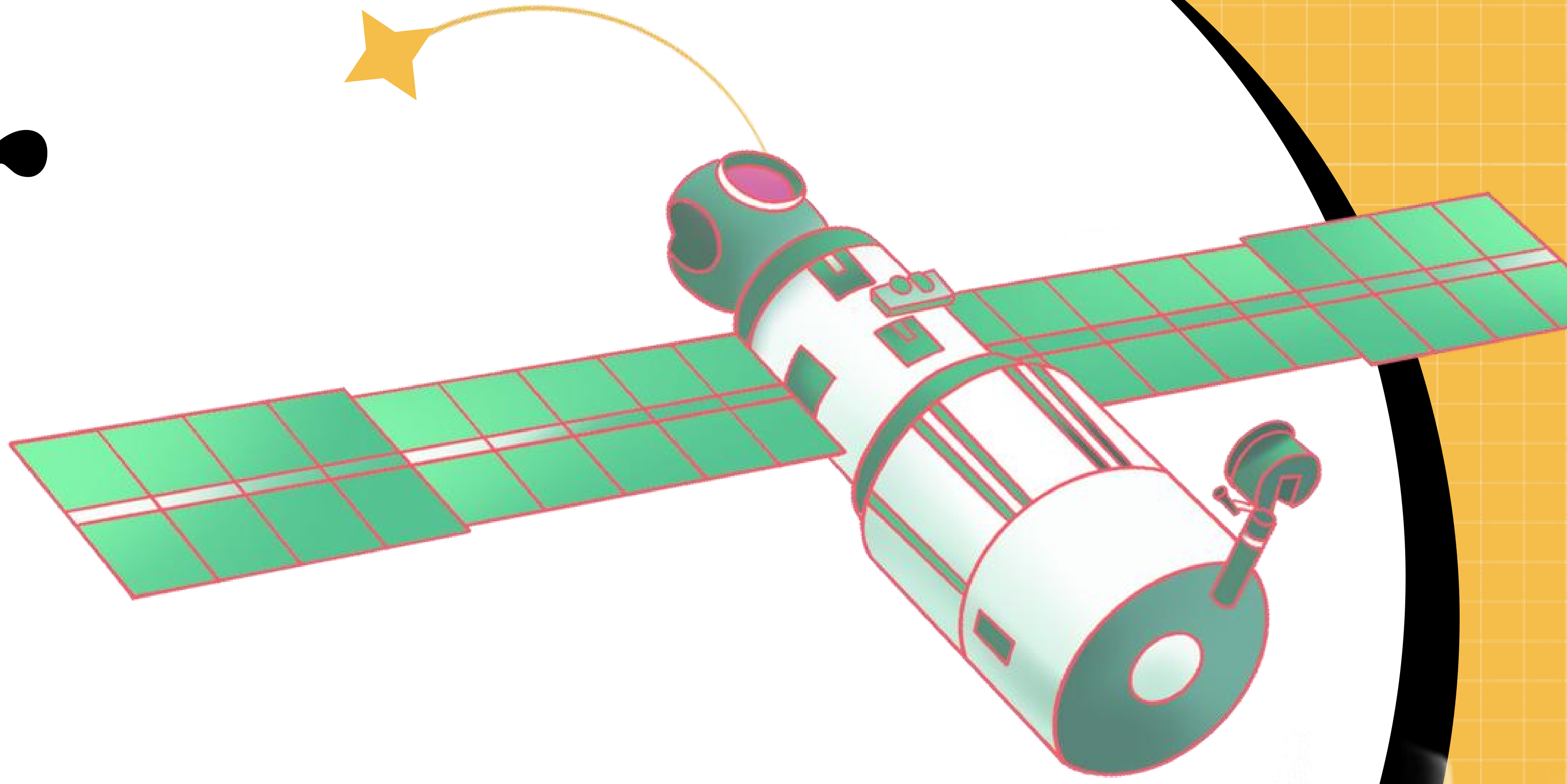
Teamwork experience

Communication and knowledge sharing

Teamwork experience

Collaboration tools

Thanks for watching



Team members:

Cao Ziqian, Zhu Hongjia, Wang Zihuo, Long Zijie, Wang Yuxin, Fang Zhengshen, Liu Shuo, Zhang Yiying, Zou Junzi, Liu Jiayi, Wang Shizhe

QMUL