**Aniruth Narayanan S**

**Kishore K G**

**Semantic Navigation of Robot with Symbolic Reasoning in ROS Simulation**

**Description**

This project aims to simulate a mobile robot capable of performing navigation tasks based on symbolic, semantic instructions instead of raw coordinate-based goals. The robot interprets human-intuitive commands such as "navigate to the room containing a chair" by querying a manually defined symbolic knowledge base. A lightweight reasoning module then translates this semantic goal into actionable coordinates, which are passed to ROS’s navigation stack to execute path planning and motion.

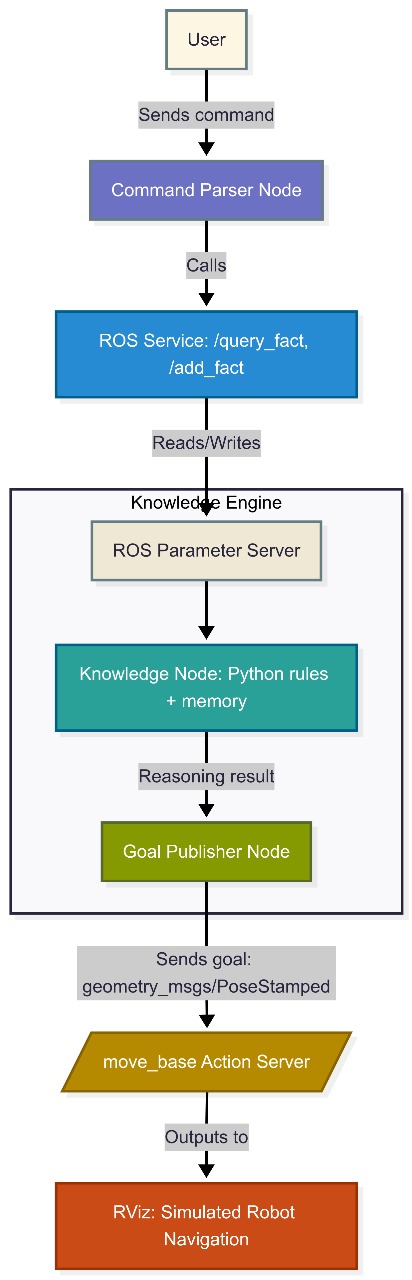
Rather than implementing complex NLP or full task planners, this project emphasizes symbolic grounding — how abstract concepts can be connected to known facts about the environment — and integrates this symbolic layer with ROS’s existing navigation tools. This type of reasoning is foundational in cognitive robotics but remains underexplored in beginner-level robotic projects.

**Use Case**

Service and maintenance robots are expected to follow high-level instructions such as “go to the kitchen” or “find the exit” .In real-world applications like healthcare robotics, facility assistance, or home automation, such commands must be grounded to physical locations or objects.

Our project replicates this concept in a simulated environment by allowing the robot to make navigation decisions using symbolic facts (e.g., “Find room with chair AND no obstacle AND not visited recently”). This enables the robot to respond to goal descriptions based on environmental context, making it adaptable to various symbolic inputs without relying on direct coordinates or complex user interfaces.

**Architecture**

****

**Outcome – Simulation**

The project will be entirely implemented in simulation. RViz will be used for 2D visualization of maps, robots, and goal states. No physical hardware will be used.

**List of Tools**

| **Component/Tool** | **Role/Function** |
| --- | --- |
| Simulated LaserScan | Obstacle detection for navigation stack |
| Odometry | Robot pose estimation |
| move\_base | Path planning and obstacle avoidance |
| amcl | Localization on static map |
| RViz | Visualization of robot, map, and goals |
| Gazebo (optional) | Physics simulation of environment |
| Python / Prolog | Symbolic reasoning and logic processing |
| YAML / Param Server | Stores environmental knowledge (e.g., room → object mapping) |

**Justification of Scope for Team Size and Evaluation**

**Complexity of the Problem**

The complexity of this project stems from bridging two different domains: symbolic reasoning (which handles human-like logic such as interpreting “go to the nearest room with a chair”) and traditional robotic navigation (which involves precise geometric movement and obstacle avoidance). While ROS supports both logic-based processing and navigation separately, their integration without complex NLP is uncommon and needs communication between a knowledge base and motion planning tools. This creates a meaningful research focused implementation that still remains feasible.

**Division of Tasks Between Members**

* **Kishore:**
  + Setup and integration of move\_base, RViz, and localization stack
  + Goal publisher node and simulation setup
* **Aniruth:**
  + Implementation of knowledge base and reasoning logic
  + Query handling and mapping symbolic goals to pose data

Both team members will jointly work on simulation testing and final integration.

**Feasibility**

* No requirement for real hardware or physical testing reduces risk.
* The project uses established ROS tools (*move\_base, RViz, tf*), allowing focus on the reasoning integration.
* The symbolic layer is kept lightweight (rule-based), avoiding the complexity of full NLP or learning models.

**Workload and Deliverables**

* Symbolic reasoning module (query + logic-based inference)
* Manual knowledge base setup for environment (e.g., YAML files)
* ROS interface for converting symbolic goal → pose
* Navigation system integration and demonstration in simulation
* All components are clearly scoped and divisible across the two-member team

**Scope Appropriateness**

This project is neither trivial nor excessive:

* Adds a reasoning layer over basic navigation, making the robot respond to high-level commands.
* Tackles an underused but important area in beginner ROS projects (semantic understanding).
* It does not require large datasets, real robots, or learning models.
* It allows clear evaluation through simulation demonstrations.
* It addresses a real-world robotics problem in an approachable, research-aligned way.