Navigation Illumination

Shedding Light on the ROS Navstack

David V. Lu!!
Robots Using ROS

Cornerstone of ROS Open Source Platform

Dozens of Supported Hardwares
Overview of ROS Navigation
Costmap Data Sources
Obstacle Inflation
Global Costmap and Plan
Social Navigation
Monolithic Costmap
Costmap Overlap
Limited Update Information
Layered Costmaps
Two Pass Update Process
Layer Combination

\[ \begin{array}{c|c|c}
\text{Layer 1} & \text{Layer 2} & \text{Result} \\
\hline
\text{Layer 3} & \text{Layer 4} & \text{Result} \\
\end{array} \]
Costmap Layers

Obstacles Layer
Inflation Layer
Static Layer
Range Sensor Layer
Proxemic Layer
Claustrophobic Layer
class Layer
{

public:
    void initialize( LayeredCostmap* parent, std::string name, tf::TransformListener *tf );

virtual void updateBounds(
    double robot_x, double robot_y, double robot_yaw,
    double* min_x, double* min_y, double* max_x, double* max_y) {}

virtual void updateCosts(Costmap2D& master_grid,
    int min_i, int min_j, int max_i, int max_j) {}
Layered Costmaps

github.com/ros-planning/navigation
github.com/wg-perception/people
github.com/DLu/navigation_layers
Overview of ROS NavStack
global_planner
BaseLocalPlanner vs. DWALocalPlanner
Weighted Sum =
oscillation_cost
+ costmap_cost
+ goal_distance_cost
+ path_distance_cost
+ goal_alignment_cost
+ path_alignment_cost
Scoring Different Trajectories

Vary x position and x velocity
Implementing a Cost Function

class TrajectoryCostFunction {

... 

virtual double scoreTrajectory(Trajectory &traj);

... 

}
MoveBase The Next Generation

MoveBase

State Machine

Global Navigator

Local Navigator
Move Base Social State Machine
DWA Plugin Planner and MoveBase2

https://github.com/DLu/navigation
/tree/groovy_plugin_planner

https://github.com/DLu/navigation
/tree/groovy_mbsplit
ROS Navigation

https://github.com/ros-planning/navigation