A highly flexible navigation framework

Move Base Flex

Sebastian Pütz
Osnabrück University
spuetz@uos.de

Jorge Santos Simón
Magazino GmbH
santos@magazino.eu
Why Move Base Flex?
...because move_base is not FLEX

- Magazino GmbH needed smarter navigation
- Osnabrück University needed extendable framework

Vs.
The Core Features

• Fully backwards-compatible with current ROS navigation
• Actions for the submodules planning, controlling and recovering, and services to query the costmaps are provided
• Comprehensive result and feedback information on all actions, including error codes and messages from the loaded plugins
• Separation between an abstract navigation framework and concrete implementations
Move Base Flex Architecture

- Actions for the submodules planning, controlling and recovering
  - Actions:
    - /get_path
    - /recovery
    - /move_base
  - Services:
    - /check_pose_cost
    - /clear_costmaps
    - /make_plan
    - /srvs/empty

- Services to query the costmaps are provided
  - Services:
    - /move_base
    - /move_base_simple/goal
    - /move_base_simple/pose
    - /move_base_msgs/Costmap

Abstract Move Base Flex Level

Move Base Implementation Level
Move Base Flex Architecture

- Actions for the submodules planning, controlling and recovering
- Fully backwards-compatible with current ROS navigation
- Services to query the costmaps are provided
Move Base Flex Architecture

- The abstract execution classes contain the main execution logic
- They are not bound to any map representation
- Thereby, the system can easily be used for other approaches
Move Base Flex Architecture

- Based on abstract execution classes
- The interfaces in the nav core inherit the abstract interfaces
- These abstract interfaces provide a richer interface
- We extended the nav_core without breaking plugin API
- The new abstract interfaces allow plugins to return valuable information
In our framework, MoveBase is just a particular implementation of a navigation system. However, we provide a SimpleNavigationServer class which does not use costmaps.
Move Base Flex mbf_2d_nav
global_planner,
local_planner
Move Base Flex mbf_simple
mesh_planner
Starting with Move Base Flex

- Old good move_base action and services are there, but...
- ...now you can use individual components: escape, query maps...
- Easy start with the SMACH tutorial in the Wiki: http://wiki.ros.org/move_base_flex/Tutorials
- More tutorials comming sonn!
- Try other executives: Scripts, Behavior Trees...

Writing a simple SMACH for Move Base Flex

Description: In this tutorial you learn how to set up a SMACH to use Move Base Flex for flexible and more specific navigation tasks. Using a SMACH lets you easily include your navigation tasks into your global robot behavior state machine. We will use RViz to receive a goal pose as the first input for the SMACH.

Keywords: smach, move_base_flex, navigation, planning, state machine

Tutorial Level: ADVANCED

1. Description

In this tutorial we address the actions GetPath, ExecPath and Recovery provided by Move Base Flex. While GetPath runs the global planner searching for a path to the target pose, ExecPath runs the local planner executing the planned path. The Recovery action can be used to execute various behaviors for error handling during planning and controlling. We connect these actions by setting up a SMACH state machine using Simple Action States. In addition to the actions described above, the implementation of a state that receives a navigation goal by the user is required. The target pose can be easily set via the visualization tool RViz and published on a specific topic.

1.1 SMACH State Machine

- SM_ROOT
- WAIT_FOR_GOAL
- Start
- Goal
- End
What to do with Move Base Flex

- Check start/goal poses before planning
- Replan only if strictly needed
- Exploit anytime planners
- Use the right recovery behavior for each error
- Introspection in the navigation decision-making
- Monitor progress and log data
- ...
Move Base Flex at Magazino

Some examples...

- avoid replanning
- check start/goal poses
- escape from collision
Avoid replanning
Future Work

- First release on October
- Grid Map (ETH Zürich, ASL: https://github.com/ethz-asl/grid_map)
- Allow multiple planners and controllers
- Select applicable plugins at runtime
- Add pause/resume interface for the controller
- Plans with waypoints
- ... <your ideas here>
A highly flexible navigation framework

Move Base Flex

wiki.ros.org/move_base_flex
github.com/magazino/move_base_flex
Join the Team – We are hiring!
For more details talk to our colleagues at the ROScon
or visit our homepage https://www.magazino.eu/job/

- Robot Deployment Engineer (m/f)
- Trainee Robotics - 18 months (m/f)
- Software QA and Testing Engineer (m/f)
- AI Research Engineer (m/f)
- AI Software Architect (m/f)
- Cloud DevOps Engineer (m/f)

- Multi-Robot Coordination Specialist (m/f)
- Robot UX and Frontend Developer (m/f)
- Robots Application Engineer (m/f)
- Senior Navigation Engineer (m/f)
- Senior Software Developer (m/f)