TTK22 Project Report

Jae Hyeong Hwang

November 2020

1 Purpose of Report

Purpose of this report is understanding DUNE by controlling ASV caravela with generated search pattern. A typical lawn mower search pattern is implemented for the task. The task should wait for the vehicle to be ready (on service mode), and then activate a FollowReference maneuver with the path. Finally if abort is received stop the mission immediately.

Figure 1: ASV Caravela

2 Approach

By following general way to command DUNE vehicles,

- Send PlanControl command to start a plan consisting of a single FollowReference maneuver
- Guide the vehicle by sending Reference messages and listening to FollowRefState and Abort messages
- If Abort is received, stop controlling the vehicle immediately

In order to send PlanControl command, we have to look into what's inside the PlanControl. It is described in figure 2. Reference messages should include waypoints that search pattern includes. FollowRefState should informations to connect vehicles and command. Abort shall be consumed by describing when it should be aborted and should include function that stops controlling vehicle.

3 Organization

3.1 namespace Test

The Task.cpp file is located at Maneuver/Test folder. using DUNE_NAMESPACES to include a plethora of other namespaces. `using std::vector;` was included to use vector later on.

3.2 struct Arguments

Variables are defined in this structure, such as current_lat which is for current latitude. All variables that is used in function are defined here.

3.3 struct Task: public DUNE::Tasks::Task

some other variables and accronym for IMC classes are defined here.

3.4 Task(const std::string name, Tasks::Context ctx):

Parameters that will be used in the task are added by using `param` command. It is for setting parameters and bind it to c++ variable. Also, IMC classes that will be consumed are binded in this function.

3.5 consume

3.5.1 Abort

It is for abort the task when the vehicle has to be stopped.

3.5.2 EstimatedState

Class `EstimatedState` is for estimating the vehicle's state. Latitude and longitude of caravela were estimated by equation following WGS84 in order to describe it in SI units. As the latitudinal and longitudinal values are described in radians. Then send the message.

3.5.3 FollowRefState

In `FollowRefState` consume, it waits for the first reference in the current position. Then, assign suitable flags then dispatch FollowRefState.

3.6 onMain

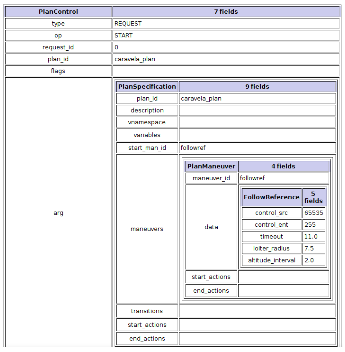
`PlanControl` will be sent consisting of a `FollowReference`, `PlanManeuver`, `PlanSpecification`. Then, will send vehicle to waypoints to perform a lawn mower search pattern.

4 Configuration

Here we will see configurations of each classes that were used.

4.1 PlanControl

PlanControl is consisted of type, op, request_id, plan_id, flags and args. A args consisted of Planspecification, which is consisted of plan_id, start_man_id and maneuvers. A Maneuvers is consisted of PlanManeuver. There are several datas needed to consist of PlanManeuver also. We can check this intuitively by figure. 2

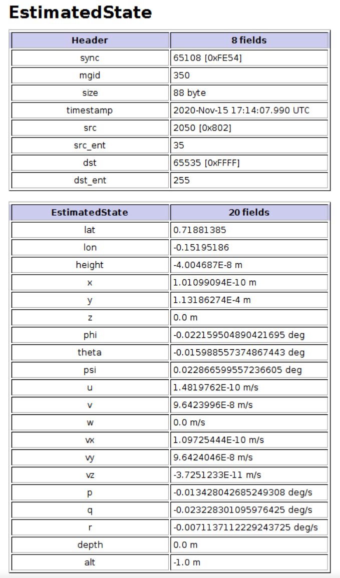


<table>
<tr><th>PlanControl</th><th>7 fields</th></tr>
<tr><td>type</td><td>REQUEST</td></tr>
<tr><td>op</td><td>START</td></tr>
<tr><td>request_id</td><td>0</td></tr>
<tr><td>plan_id</td><td>caravela_plan</td></tr>
<tr><td>flags</td><td></td></tr>
<tr><td>arg</td><td>
<table>
<tr><th>PlanSpecification</th><th>9 fields</th></tr>
<tr><td>plan_id</td><td>caravela_plan</td></tr>
<tr><td>description</td><td></td></tr>
<tr><td>vnamespace</td><td></td></tr>
<tr><td>variables</td><td></td></tr>
<tr><td>start_man_id</td><td>followref</td></tr>
<tr><td>maneuvers</td><td>
<table>
<tr><th>PlanManeuver</th><th>4 fields</th></tr>
<tr><td>maneuver_id</td><td>followref</td></tr>
<tr><td>data</td><td>
<table>
<tr><th>FollowReference</th><th>5 fields</th></tr>
<tr><td>control_src</td><td>65535</td></tr>
<tr><td>control_ent</td><td>255</td></tr>
<tr><td>timeout</td><td>11.0</td></tr>
<tr><td>loiter_radius</td><td>7.5</td></tr>
<tr><td>altitude_interval</td><td>2.0</td></tr>
</table>
</td></tr>
<tr><td>start_actions</td><td></td></tr>
<tr><td>end_actions</td><td></td></tr>
</table>
</td></tr>
<tr><td>transitions</td><td></td></tr>
<tr><td>start_actions</td><td></td></tr>
<tr><td>end_actions</td><td></td></tr>
</table>
</td></tr>
</table>

Figure 2: PlanControl configuration

4.2 EstimatedState

In this task, only `lat`, `lon`, `x`, `y` were used to estimate the position of vehicle. Lattitudinal and longitudinal values could be measured but in radians. So had to traslate it into SI units by getting the error value of x and y.



EstimatedState

Header	8 fields
sync	65108 [0xFE54]
mgid	350
size	88 byte
timestamp	2020-Nov-15 17:14:07.990 UTC
src	2050 [0x802]
src_ent	35
dst	65535 [0xFFFF]
dst_ent	255

EstimatedState	20 fields
lat	0.71881385
lon	-0.15195186
height	-4.004687E-8 m
x	1.01099094E-10 m
y	1.13186274E-4 m
z	0.0 m
phi	-0.022159504890421695 deg
theta	-0.015988557374867443 deg
psi	0.022866599557236605 deg
u	1.4819762E-10 m/s
v	9.6423996E-8 m/s
w	0.0 m/s
vx	1.09725444E-10 m/s
vy	9.6424046E-8 m/s
vz	-3.7251233E-11 m/s
p	-0.013428042685249308 deg/s
q	-0.023228301095976425 deg/s
r	-0.0071137112229243725 deg/s
depth	0.0 m
alt	-1.0 m

Figure 3: EstimatedState configuration

4.3 FollowRefState

control_src is the identifier of source system. ”0xFFF” is used to allow for any system s to command references.[1] control_ent is the identifier of the entity that is allowed to control the vehicle. “0xFF” is used, to allow for any entity command. Proximity was set as same as `XY_NEAR` to make it near the horizontal plane. State was set as same as `FR_WAIT` to wait for the first reference.

5 Results

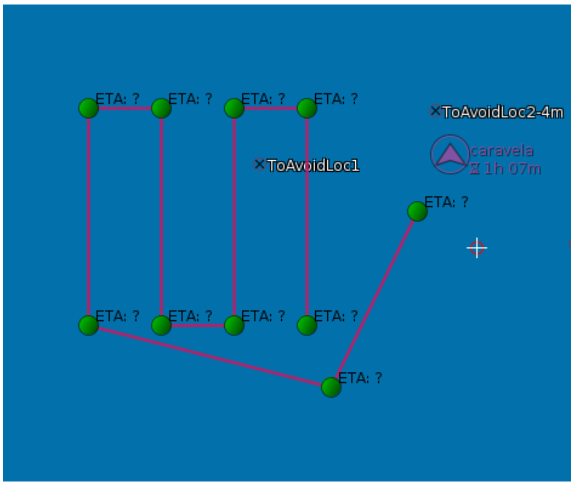
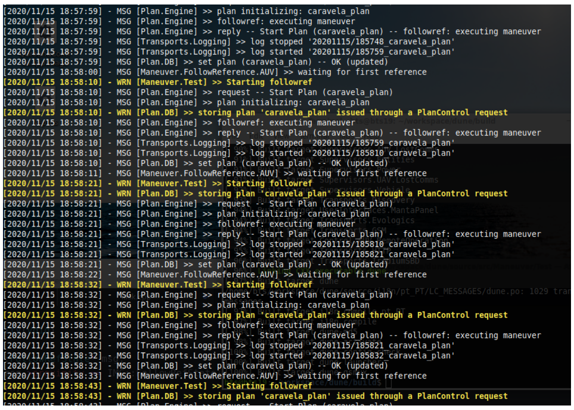


Figure 4: Screenshot in Neptus

Unfortunately, the code didn't work enough well as we can see in figure 4. It was showing the waypoints that was intended, as lawn mower search pattern. However, the vehicle didn't perform the task. Couldn't figure out the exact reasons but we can guess refering to figure 5.



```
[2020/11/15 18:57:59] - MSG [Plan.Engine] >> plan initializing: caravela_plan
[2020/11/15 18:57:59] - MSG [Plan.Engine] >> followref: executing maneuver
[2020/11/15 18:57:59] - MSG [Plan.Engine] >> reply -- Start Plan (caravela_plan) -- followref: executing maneuver
[2020/11/15 18:57:59] - MSG [Transports.Logging] >> log stopped '20201115/185748_caravela_plan'
[2020/11/15 18:57:59] - MSG [Transports.Logging] >> log started '20201115/185759_caravela_plan'
[2020/11/15 18:57:59] - MSG [Plan.DB] >> set plan (caravela_plan) -- OK (updated)
[2020/11/15 18:58:00] - MSG [Maneuver.FollowReference.AUV] >> waiting for first reference
[2020/11/15 18:58:10] - WRN [Maneuver.Test] >> Starting followref
[2020/11/15 18:58:10] - MSG [Plan.Engine] >> request -- Start Plan (caravela_plan)
[2020/11/15 18:58:10] - MSG [Plan.Engine] >> plan initializing: caravela_plan
[2020/11/15 18:58:10] - WRN [Plan.DB] >> storing plan 'caravela_plan' issued through a PlanControl request
[2020/11/15 18:58:10] - MSG [Plan.Engine] >> followref: executing maneuver
[2020/11/15 18:58:10] - MSG [Plan.Engine] >> reply -- Start Plan (caravela_plan) -- followref: executing maneuver
[2020/11/15 18:58:10] - MSG [Transports.Logging] >> log stopped '20201115/185759_caravela_plan'
[2020/11/15 18:58:10] - MSG [Transports.Logging] >> log started '20201115/185810_caravela_plan'
[2020/11/15 18:58:10] - MSG [Plan.DB] >> set plan (caravela_plan) -- OK (updated)
[2020/11/15 18:58:11] - MSG [Maneuver.FollowReference.AUV] >> waiting for first reference
[2020/11/15 18:58:21] - WRN [Maneuver.Test] >> Starting followref
[2020/11/15 18:58:21] - WRN [Plan.DB] >> storing plan 'caravela_plan' issued through a PlanControl request
[2020/11/15 18:58:21] - MSG [Plan.Engine] >> request -- Start Plan (caravela_plan)
[2020/11/15 18:58:21] - MSG [Plan.Engine] >> plan initializing: caravela_plan
[2020/11/15 18:58:21] - MSG [Plan.Engine] >> followref: executing maneuver
[2020/11/15 18:58:21] - MSG [Plan.Engine] >> reply -- Start Plan (caravela_plan) -- followref: executing maneuver
[2020/11/15 18:58:21] - MSG [Transports.Logging] >> log stopped '20201115/185810_caravela_plan'
[2020/11/15 18:58:21] - MSG [Transports.Logging] >> log started '20201115/185821_caravela_plan'
[2020/11/15 18:58:21] - MSG [Plan.DB] >> set plan (caravela_plan) -- OK (updated)
[2020/11/15 18:58:22] - MSG [Maneuver.FollowReference.AUV] >> waiting for first reference
[2020/11/15 18:58:32] - WRN [Maneuver.Test] >> Starting followref
[2020/11/15 18:58:32] - MSG [Plan.Engine] >> request -- Start Plan (caravela_plan)
[2020/11/15 18:58:32] - MSG [Plan.Engine] >> plan initializing: caravela_plan
[2020/11/15 18:58:32] - WRN [Plan.DB] >> storing plan 'caravela_plan' issued through a PlanControl request
[2020/11/15 18:58:32] - MSG [Plan.Engine] >> followref: executing maneuver
[2020/11/15 18:58:32] - MSG [Plan.Engine] >> reply -- Start Plan (caravela_plan) -- followref: executing maneuver
[2020/11/15 18:58:32] - MSG [Transports.Logging] >> log stopped '20201115/185821_caravela_plan'
[2020/11/15 18:58:32] - MSG [Transports.Logging] >> log started '20201115/185832_caravela_plan'
[2020/11/15 18:58:32] - MSG [Plan.DB] >> set plan (caravela_plan) -- OK (updated)
[2020/11/15 18:58:33] - MSG [Maneuver.FollowReference.AUV] >> waiting for first reference
[2020/11/15 18:58:43] - WRN [Maneuver.Test] >> Starting followref
[2020/11/15 18:58:43] - WRN [Plan.DB] >> storing plan 'caravela_plan' issued through a PlanControl request
[2020/11/15 18:58:43] - MSG [Plan.Engine] >> request -- Start Plan (caravela_plan)
```

Figure 5: Simulation results

The code was compiled successfully and seemed to be running but the problem was reference point was not sent successfully. So after simulating caravela.ini it was repeating inside the `onMain` function and just sending the sign ("Starting followref")

6 Discussion

Through this small project, I could learn a lot about DUNE system and how to write the code, also could understand how DUNE system works. However, it took some time to get an overview of my tasks and understanding of overall system which resulted in poor performance.
Future task would be successfully implement the code and make the vehicle perform the task properly, also applying it to Neptus.

References

[1] U. do Porto Faculdade de Engenharia LSTS. Maneuvering messages. [Online]. Available: https://www.lsts.pt/docs/imc/imc-5.4.11/Maneuvering.html # follow-reference-state