bag_analysis

April 26, 2022

1 Bag Analysis Task

1.0.1 Import

[1]:
```
import bagpy
from bagpy import bagreader
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
```

1.0.2 Read ROS BAG file using bagpy

[2]:
```
PATH = '/home/nayan/NODE_Robotics/environment_files'

# bag1: bag file from 01.10.2020 between 11:20-11:30
# bag2: random bag file for reference
bag1_name = 'dummy_env-agv-50231.
 ↪agv-2020-10-01T082312+0200_2020-10-01-11-28-17_37.bag'
bag2_name = 'dummy_env-02LSTRV30209OA001.
 ↪agv-2020-09-17T101249+0200_2020-09-18-15-07-52_347.bag'

b1 = bagreader(PATH + '/' + bag1_name)
b2 = bagreader(PATH + '/' + bag2_name)
```

```
[INFO]  Data folder /home/nayan/NODE_Robotics/environment_files/dummy_env-
agv-50231.agv-2020-10-01T082312+0200_2020-10-01-11-28-17_37 already exists. Not
creating.
[INFO]  Data folder /home/nayan/NODE_Robotics/environment_files/dummy_env-02LSTR
V30209OA001.agv-2020-09-17T101249+0200_2020-09-18-15-07-52_347 already exists.
Not creating.
```

1.1 For bag1

1.1.1 Inspect Topics

[3]:
```
b1.topic_table
```

[3]:
```
                                    Topics  \
0                                 /BMS/SOC
1                                   /BMS/U
2    /bmw_control/left/motor_state_throttled
3   /bmw_control/right/motor_state_throttled
4                                   /clock
..                                    …
67                        /stop_signal_soft
68        /stop_signal_soft_teleop_override
69                                    /tf
70                               /tf_static
71                /vehicle_status_throttled

                                    Types  Message Count  Frequency
0                         std_msgs/Float64           3000   9.924763
1                         std_msgs/Float64           3000   9.924786
2   knx_motor_control/WittensteinMotorState           2888   9.920608
3   knx_motor_control/WittensteinMotorState           2888   9.921171
4                      rosgraph_msgs/Clock           6065  19.861604
..                                     …            …        …
67                           std_msgs/Bool              2   0.022214
68                           std_msgs/Bool              3   0.023870
69                     tf2_msgs/TFMessage          59442        NaN
70                     tf2_msgs/TFMessage              1        NaN
71               knx_msgs/VehicleStatusMsg            231   0.682104

[72 rows x 4 columns]
```

1.1.2 Inspect Topic: /cmd_vel to check when robot stops

[4]:
```
cmd_vel_msg = b1.message_by_topic('/cmd_vel')
df_cmd_vel_msg = pd.read_csv(cmd_vel_msg)
df_cmd_vel_msg
```

[4]:
```
              Time  linear.x  linear.y  linear.z  angular.x  angular.y  \
0     1.601544e+09  0.419721       0.0       0.0        0.0        0.0
1     1.601544e+09  0.427854       0.0       0.0        0.0        0.0
2     1.601544e+09  0.427854       0.0       0.0        0.0        0.0
3     1.601544e+09  0.429113       0.0       0.0        0.0        0.0
4     1.601544e+09  0.430095       0.0       0.0        0.0        0.0
…              …         …        …         …          …          …
8995  1.601545e+09  0.000000       0.0       0.0        0.0        0.0
```

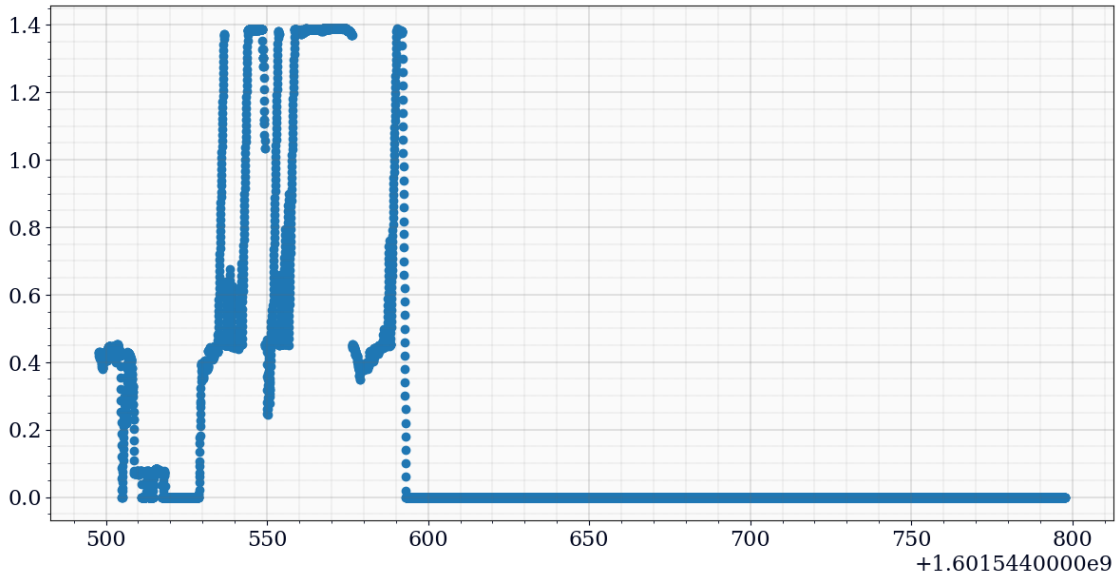
```
8996  1.601545e+09  0.000000       0.0       0.0       0.0       0.0
8997  1.601545e+09  0.000000       0.0       0.0       0.0       0.0
8998  1.601545e+09  0.000000       0.0       0.0       0.0       0.0
8999  1.601545e+09  0.000000       0.0       0.0       0.0       0.0
```

```
      angular.z
0      0.141330
1      0.112794
2      0.112794
3      0.108376
4      0.104930
…           …
8995   0.000000
8996   0.000000
8997   0.000000
8998   0.000000
8999   0.000000

[9000 rows x 7 columns]
```

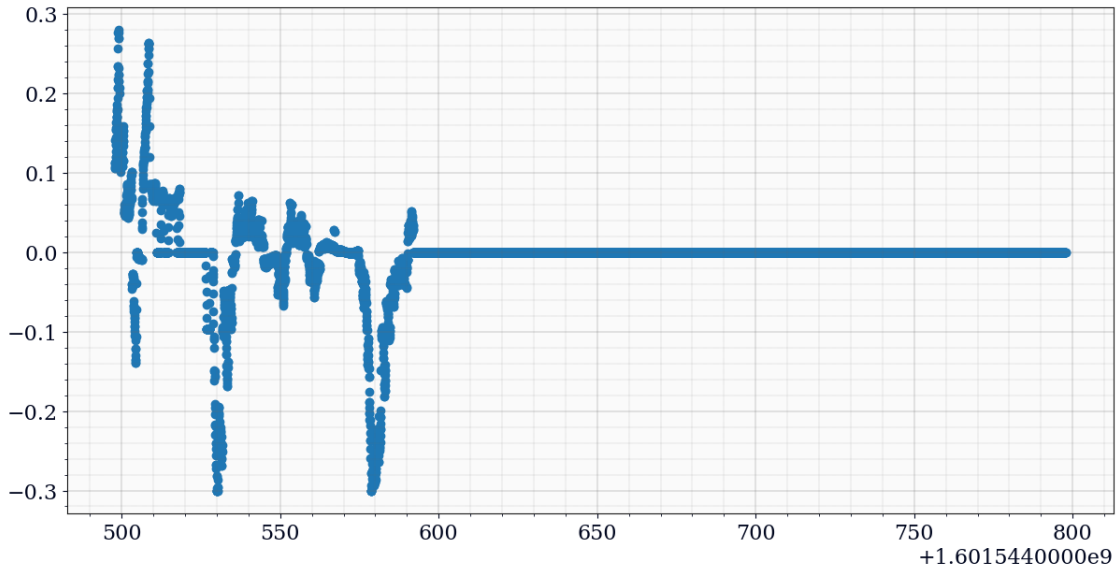
1.1.3 Plotting time vs linear velocity in x graph

[5]:
```
fig, ax = bagpy.create_fig(1)
ax[0].scatter(x='Time', y='linear.x', data=df_cmd_vel_msg)
plt.show()
```



1.1.4 Plotting time vs angular velocity in z graph

[6]:
```
fig, ax = bagpy.create_fig(1)
ax[0].scatter(x='Time', y='angular.z', data=df_cmd_vel_msg)
plt.show()
```



1.2 For bag2

1.2.1 Inspect Topics

[7]:
```
b2.topic_table
```

[7]:

	Topics
0	/BMS/SOC
1	/BMS/U
2	/bmwstr_zone_model/event_feedback
3	/bmwstr_zone_model/trigger_events
4	/bmwstr_zone_model/zone_manager_node/entry_zon…
5	/bmwstr_zone_model/zone_manager_node/exit_zone…
6	/cmd_vel
7	/cmd_vel_safety
8	/cmd_vel_safety_limited
9	/cmd_vel_safety_planned
10	/enable_gap
11	/get_input_pins_state_throttled
12	/interaction_manager/repause_monitor/cancel
13	/interaction_manager/repause_monitor/goal

```
/interaction_manager/repause_monitor/result
/interaction_manager/repause_monitor/status
   /interaction_manager/wait_for_lease/goal
 /interaction_manager/wait_for_lease/result
 /interaction_manager/wait_for_lease/status
                                      /ipa_log
                                /knx_monitoring
                               /laser_front/scan
                                /laser_rear/scan
                     /long_term_slam/lts_status
                      /long_term_slam/particles
                         /lts_confidence_marker
                                  /magnet_sensor
                                /mission_active
              /move_base/EbandLocalPlanner/eband
 /move_base/EbandLocalPlanner/eband_bubbles
    /move_base/EbandLocalPlanner/local_plan
               /move_base/LatticePlanner/plan
                       /move_base/current_goal
                    /move_base/exe_path/cancel
                  /move_base/exe_path/feedback
                      /move_base/exe_path/goal
                    /move_base/exe_path/result
                    /move_base/exe_path/status
                      /move_base/get_path/goal
                    /move_base/get_path/result
                    /move_base/get_path/status
                                 /move_base/goal
            /move_base/local_costmap/costmap
  /move_base/local_costmap/costmap_updates
        /move_base/local_costmap/footprint
                /move_base/move_base/status
                 /move_base/recovery/status
                           /move_base/result
                           /move_base/status
  /obstacle_collision_filter/obstacle_zone
                                       /odom
                                  /park_mode
                                      /queue
                                /queue/status
                   /state_machine_event_log
                                 /stop_signal
                /stop_signal_from_flexisoft
 /stop_signal_from_ultrasonic_stop_control
                                         /tf
                   /vehicle_status_throttled
```

	Types	Message Count	\
0	std_msgs/Float64	2456	
1	std_msgs/Float64	2456	
2	std_msgs/String	5	
3	std_msgs/String	14	
4	std_msgs/String	5	
5	std_msgs/String	6	
6	geometry_msgs/Twist	9000	
7	geometry_msgs/Twist	4990	
8	geometry_msgs/Twist	4990	
9	geometry_msgs/Twist	4990	
10	std_msgs/Bool	600	
11	knx_wago_io/IOStatePublisher	2897	
12	actionlib_msgs/GoalID	8	
13	commander_bmw/MonitorRepauseActionGoal	14	
14	bmwstr_interaction_zone_manager/MonitorRepause…	16	
15	actionlib_msgs/GoalStatusArray	1538	
16	bmwstr_interaction_zone_manager/WaitForLeaseAc…	8	
17	bmwstr_interaction_zone_manager/WaitForLeaseAc…	8	
18	actionlib_msgs/GoalStatusArray	1516	
19	diagnostic_msgs/DiagnosticArray	7903	
20	std_msgs/String	5922	
21	sensor_msgs/LaserScan	7645	
22	sensor_msgs/LaserScan	7647	
23	ipa_navigation_msgs/LTSStatus	300	
24	visualization_msgs/MarkerArray	1499	
25	visualization_msgs/MarkerArray	300	
26	std_msgs/Int32MultiArray	2899	
27	std_msgs/Bool	300	
28	ipa_navigation_msgs/EBand2	4999	
29	visualization_msgs/MarkerArray	5007	
30	nav_msgs/Path	4999	
31	nav_msgs/Path	4	
32	geometry_msgs/PoseStamped	4	
33	actionlib_msgs/GoalID	5	
34	mbf_msgs/ExePathActionFeedback	4990	
35	mbf_msgs/ExePathActionGoal	9	
36	mbf_msgs/ExePathActionResult	8	
37	actionlib_msgs/GoalStatusArray	1498	
38	mbf_msgs/GetPathActionGoal	4	
39	mbf_msgs/GetPathActionResult	4	
40	actionlib_msgs/GoalStatusArray	1484	
41	commander_bmw/MoveBaseActionGoal	4	
42	nav_msgs/OccupancyGrid	372	
43	map_msgs/OccupancyGridUpdate	128	
44	geometry_msgs/PolygonStamped	1500	
45	actionlib_msgs/GoalStatusArray	1476	

46	actionlib_msgs/GoalStatusArray	1476
47	ipa_navigation_msgs/MoveBaseActionResult	3
48	actionlib_msgs/GoalStatusArray	1507
49	visualization_msgs/MarkerArray	4990
50	nav_msgs/Odometry	29939
51	std_msgs/Bool	600
52	std_msgs/String	8
53	knx_msgs/CommandQueueStatus	30
54	std_msgs/String	353
55	std_msgs/Bool	10
56	std_msgs/Bool	44998
57	std_msgs/Bool	3000
58	tf2_msgs/TFMessage	93142
59	knx_msgs/VehicleStatusMsg	231

	Frequency
0	9.997650
1	9.997507
2	0.025556
3	0.059536
4	0.026319
5	0.027398
6	29.995523
7	19.908883
8	19.912380
9	19.923352
10	1.999995
11	9.626024
12	0.067738
13	0.052430
14	0.080227
15	5.001072
16	0.038852
17	0.037597
18	5.000046
19	55.179498
20	847.847989
21	22.726642
22	22.725041
23	0.999913
24	4.998521
25	0.999936
26	9.661356
27	1.000011
28	19.931258
29	19.925056
30	19.925009

```
31      0.008686
32      0.008685
33      0.025560
34     19.915406
35      0.031919
36      0.027670
37      5.000702
38      0.008685
39      0.008686
40      5.000273
41      0.008684
42      1.666539
43      1.665340
44      4.999826
45      4.999838
46      4.999999
47      0.013502
48      5.000112
49     19.921933
50  87381.333333
51      1.999968
52      0.160036
53     43.455734
54   1532.445744
55      0.761588
56    150.064544
57      9.999580
58    397.263118
59      0.668310
```

1.2.2 Inspect Topic: /cmd_vel to check when robot stops

[8]:
```
cmd_vel_msg = b2.message_by_topic('/cmd_vel')
df_cmd_vel_msg = pd.read_csv(cmd_vel_msg)
df_cmd_vel_msg
```

[8]:

	Time	linear.x	linear.y	linear.z	angular.x	angular.y	\
0	1.600434e+09	0.167555	0.0	0.0	0.0	0.0	
1	1.600434e+09	0.167555	0.0	0.0	0.0	0.0	
2	1.600434e+09	0.168142	0.0	0.0	0.0	0.0	
3	1.600434e+09	0.168142	0.0	0.0	0.0	0.0	
4	1.600434e+09	0.168742	0.0	0.0	0.0	0.0	
…	…	…	…	…	…	…	
8995	1.600435e+09	0.142556	0.0	0.0	0.0	0.0	
8996	1.600435e+09	0.138511	0.0	0.0	0.0	0.0	
8997	1.600435e+09	0.138511	0.0	0.0	0.0	0.0	
8998	1.600435e+09	0.135267	0.0	0.0	0.0	0.0	

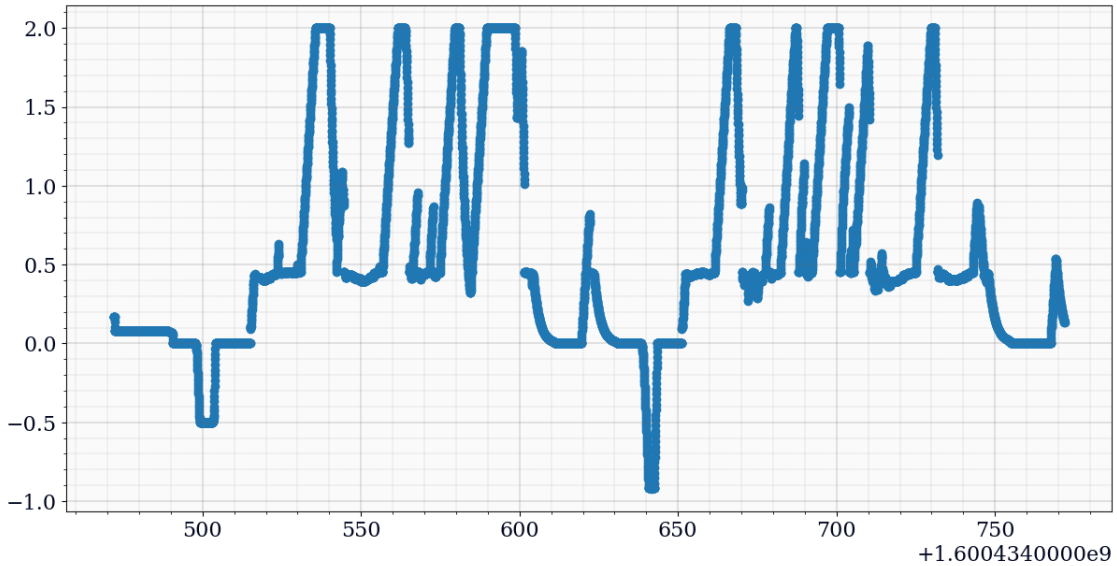
```
8999  1.600435e+09  0.131734       0.0       0.0       0.0       0.0

      angular.z
0      0.002086
1     -0.000178
2      0.001865
3     -0.000404
4     -0.000700
…           …
8995   0.002491
8996   0.002592
8997   0.002592
8998   0.003803
8999   0.002846

[9000 rows x 7 columns]
```

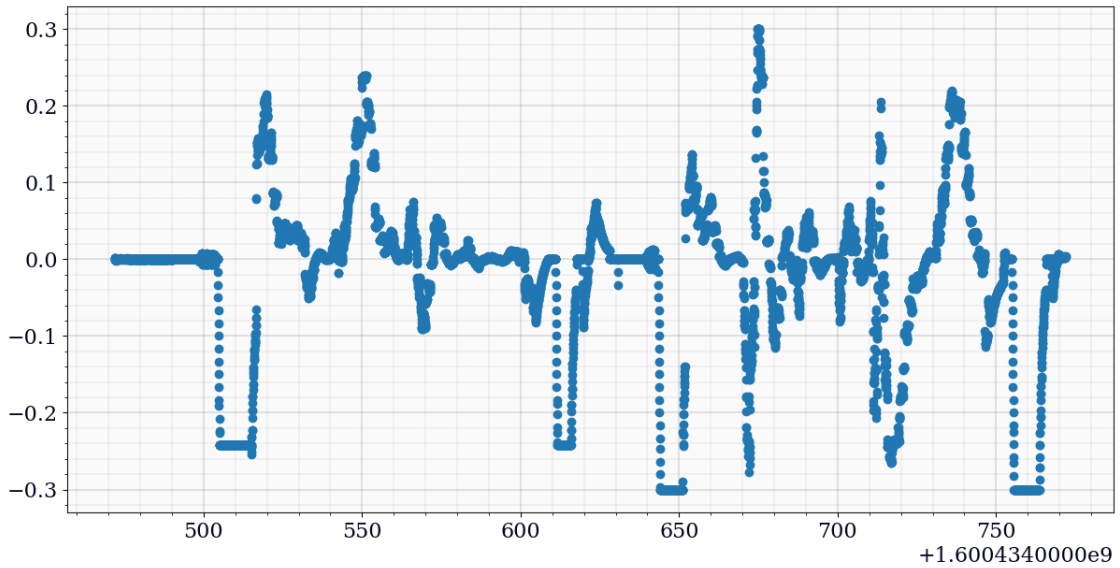
1.2.3 Plotting time vs linear velocity in x graph

[9]:
```
fig, ax = bagpy.create_fig(1)
ax[0].scatter(x='Time', y='linear.x', data=df_cmd_vel_msg)
plt.show()
```



1.2.4 Plotting time vs angular velocity in z graph

[10]:
```
fig, ax = bagpy.create_fig(1)
ax[0].scatter(x='Time', y='angular.z', data=df_cmd_vel_msg)
plt.show()
```



---

2 Analysis

Two bag files are analysed. Bag 1 is the bag file from 01.10.2020 between 11:20-11:30. Bag 2 is a random bag file. We observe the linear velocity in the x-axis and angular velocity in the z-axis of the robot using the '/cmd_vel' topic.

We notice that there is a sudden cut-off in the linear velocity in x-axis and angular velocity in z-axis in Bag 1. This indicates that the robot stopped moving and that something might be wrong in the robot.

A detailed analysis and evaluation is done in the **node_rosbag_analysis_documentation.pdf** pdf in the documentation folder.