Heterogeneous Multi Robot Integration

Siddharth Bhurat, sbhurat@umich.edu

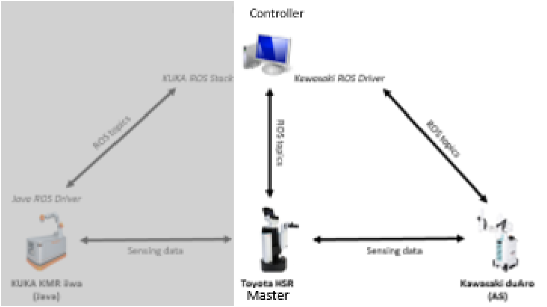


Fig. 1. Current Progress.

I. Motivation

This project aims to integrate the Toyota HSR Robot with DuAro and Kuka KMR Robot over the wifi network using ROS. Since Kuka KMR was not operational, I was mainly focusing on integrating Toyota HSR and DuAro. This enables having a unified control over all the robots from a single master and having shared sensing which helps in reducing redundant sensors. We were able to integrate ROS on DuAro and Toyota HSR with later as master and brought them to work on same ROS network. We also used april tags to find the initial transform between immobile DuAro and mobile Toyota HSR for the purpose of correct visualization in RViz. A few difficulties were encountered to make the Toyota HSR, a slave of the main controller after trying a method different than the previously implemented methods. But we were able to achieve multi-master configuration between Toyota HSR and laptop without installing any packages on Toyota HSR robot.

II. Problem Statement and Objectives

A. Review of Previous Work

Previously, they were able to integrate ROS on DuAro and Kuka KMR with a laptop as an external master but Toyota HSR was not integrated in this configuration, use april tags to perform calibration and visualize the point cloud. They also visualized both the robots on the same network using 2 different RViz windows.

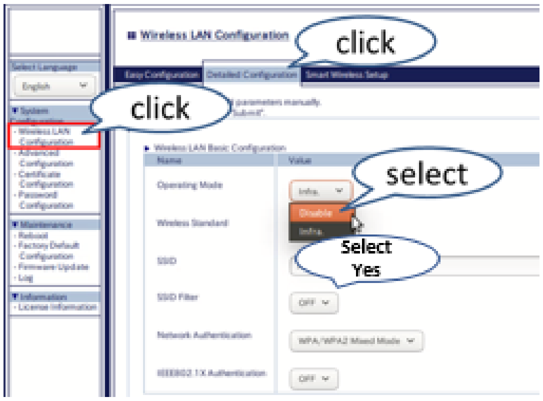
B. Planned work on this Project

Based on the review of previously done work on this project, we planned on trying other methods to configure Toyota HSR to an external master, then configure both Toyota HSR and DuAro on the same network with either laptop or Toyota HSR as the master. After that bring both the robots on the same visualization window of RViz using a single launch file, then find the transform between Toyota HSR and DuAro using april tags and visualize them in the RViz using corrected transforms.

III. Technical Approach

A. Configuring Toyota HSR

We started with configuring Toyota HSR robot, connecting it to the Lab Wifi network. But there were some issues which we faced, wherein the wifi gateway of the robot never showed the Lab Wifi network because there were some redundant wifi networks showing up in the list. After going through the documentation, we figured out that you need to go to detailed configuration section and set the SSID filter to yes as described in figure 2. This helps in filtering all the unwanted networks and keeps the one which is mentioned in the configuration. Using this we were able to connect the robot to wifi.



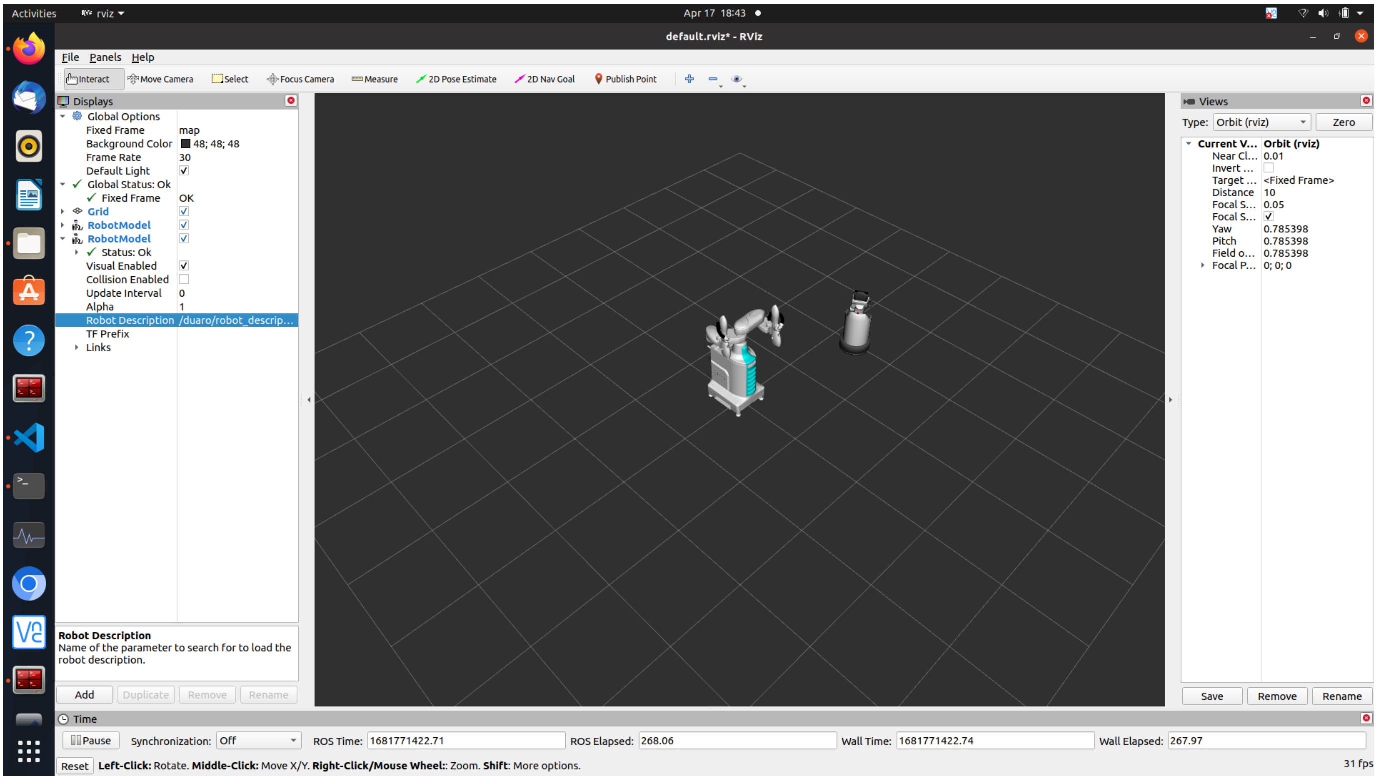
Operating Mode: Infra.
Wireless Standard
SSID
SSID Filter: OFF
Network Authentication: WPA/WPA2 Mixed Mode
IEEE802.1X Authentication: OFF

Fig. 2. Wifi Configuration.

After setting up the wifi, there was another issue with Toyota HSR, that the ROS framework of the robot was not starting up and there were no control topics available to control the robot. After removing ”ROS_MASTER_URI” environment variable, which we suspect was added to make an external laptop the ”Master” of Toyota HSR robot. On removing this variable and restarting the robot, we were able to make the initialization of the robot happen correctly and ROS runs with all the controls available to the user.

B. Configuring DuAro with Toyota HSR as Master

The first step towards configuring DuAro with Toyota HSR was to make Toyota HSR, the master of DuAro. The only step we need to do is to either change the ”ROS_MASTER_URI” to ip-address of Toyota HSR or change the terminal mode to ”hsrb_mode”, which indirectly changes the master as Toyota HSR. On running existing launch files with this configuration, there were few related to parameter and topic duplication



Global Options
Fixed Frame map
Background Color 48; 48; 48
Frame Rate 30
Default Light ☑
Global Status: Ok
✓ Fixed Frame OK
Grid ☑
RobotModel ☑
RobotModel ☑
✓ Status: Ok
Visual Enabled ☑
Collision Enabled □
Update Interval 0
Alpha 1
Robot Description /duaro/robot_descrip...
TF Prefix
Links

Robot Description
Name of the parameter to search for to load the robot description.

Type: Orbit (rviz)
Current V... Orbit (rviz)
Near Cl... 0.01
Invert ... □
Target ... <Fixed Frame>
Distance 10
Focal S... 0.05
Focal S... ☑
Yaw 0.785398
Pitch 0.785398
Field o... 0.785398
Focal P... 0; 0; 0

Synchronization: Off ROS Time: 1681771422.71 ROS Elapsed: 268.06 Wall Time: 1681771422.74 Wall Elapsed: 267.97

Left-Click: Rotate. **Middle-Click:** Move X/Y. **Right-Click/Mouse Wheel:**: Zoom. **Shift:** More options.

31 fps

Fig. 3. DuAro and HSR visualized in same RViz window

errors . The solution was to add a name-space to all the conflicting parameters and topics between these 2 robots. So, there are some parameters like ”/robot_description” which are same for both Toyota HSR and DuAro, we can add name-space to DuAro robot's launch file parameters by using the opening tag of namespace <group ns=”r1”> and a closing tag </group> for the node or parameter which requires different namespace.

/map

/robot_description	/robot_description
/robot_state_publisher	/robot_state_publisher
/jt1_vel	/jt1_vel
/base_link	/base_link

Fig. 4. Incorrect namespace which cause conflicts (Parameters on left are of HSR and parameters on the right are of DuAro)

/map

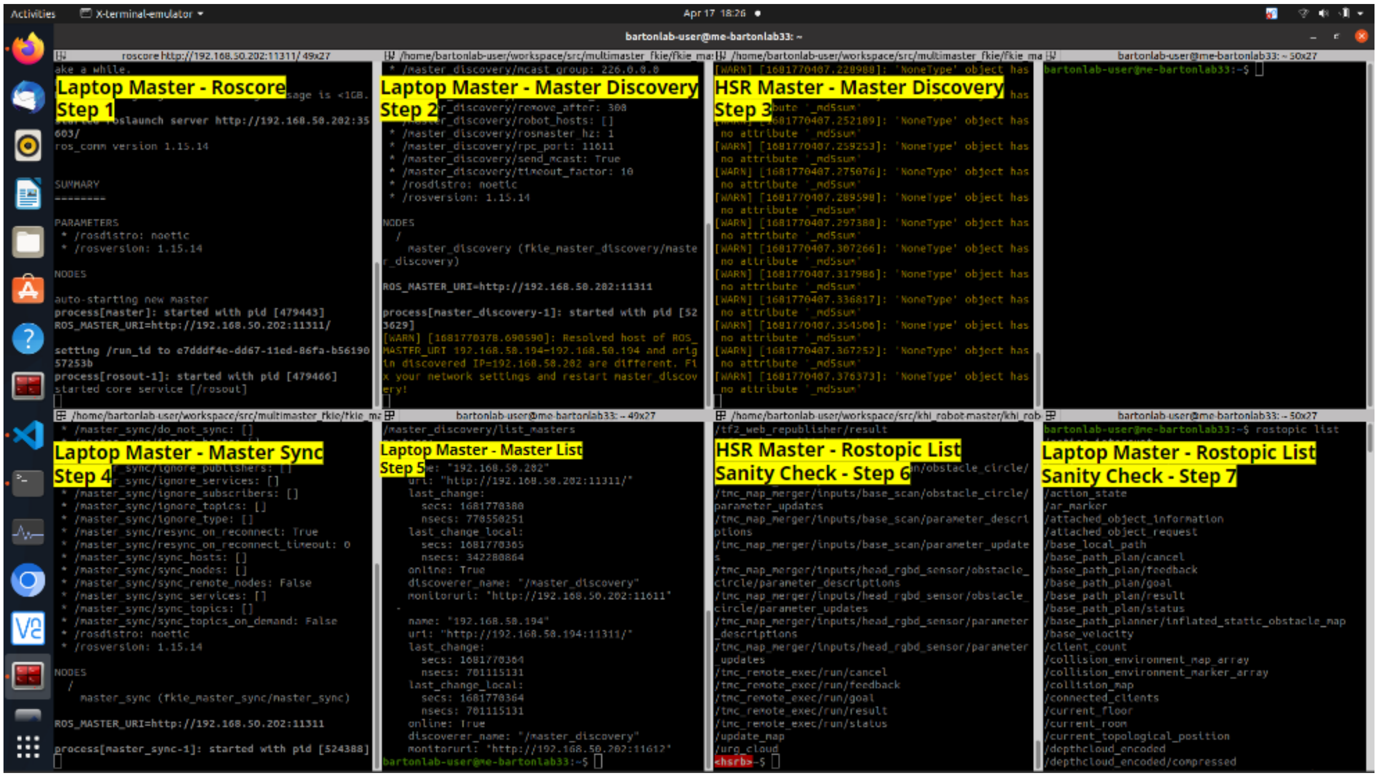
/hsr/robot_description	/duaro/robot_description
/hsr/robot_state_publisher	/duaro/robot_state_publisher
/hsr/jt1_vel	/duaro/jt1_vel
/hsr/base_link	/duaro/base_link

Fig. 5. Correction of namespaces in the DuAro Bringup launch file

This solution worked perfectly fine on implementing. But DuAro's velocity limits were being crossed when used with Toyota HSR as the master and DuAro gives the error of ”RTC Switch turned OFF” because of velocity limits being crossed. On debugging the error message, we could resolve this by interpreting the error which had the status information which is if zero, it means the velocity limits are crossed and RTC switch will be turned off for safety reasons. So, we figured out that the solution for this was to reduce the actual velocity limits of the robot joints itself, so that the velocity commands published are well below the actual velocity limits and this solved the problem and there were no more RTC switch errors. Now both the robots are well configured to work with each other using one launch file.

C. Fixing the Relative Transforms between HSR and DuAro

Now that both the robots are configured, the next step is to have correct relative transforms between the mobile Toyota HSR robot and immobile DuAro robot. For this we had to reconfigure the transform tree [1] by publishing a transform from map to Toyota HSR and from map to DuAro and then a relative transform between Toyota HSR and fixed DuAro, will make the complete environment in the RViz as good as in the real world. Then used the April Tags to find the relative transform between DuAro and Toyota HSR robot and this can be further used for SLAM in the same environment. For the april_tag ros package to work with Toyota HSR, we had to write the synchronization script which runs the the synchronization between image and the camera info coming from HSR. We had to do this because both image and camera_info were not synchronized and were being published



roscore http://192.168.50.202:11311/ 49x27

Laptop Master - Roscore
Step 1

```
ake a while.
                                          sage is <1GB.
          oslaunch server http://192.168.50.202:35
603/
ros_comm version 1.15.14


SUMMARY
========

PARAMETERS
 * /rosdistro: noetic
 * /rosversion: 1.15.14

NODES

auto-starting new master
process[master]: started with pid [479443]
ROS_MASTER_URI=http://192.168.50.202:11311/

setting /run_id to e7dddf4e-dd67-11ed-86fa-b56190
57253b
process[rosout-1]: started with pid [479466]
started core service [/rosout]
```

/home/bartonlab-user/workspace/src/multimaster_fkie/fkie_ma

Laptop Master - Master Discovery
Step 2

```
 * /master_discovery/mcast_group: 226.0.0.0
      r_discovery/remove_after: 300
      r_discovery/robot_hosts: []
 * /master_discovery/rosmaster_hz: 1
 * /master_discovery/rpc_port: 11611
 * /master_discovery/send_mcast: True
 * /master_discovery/timeout_factor: 10
 * /rosdistro: noetic
 * /rosversion: 1.15.14

NODES
  /
    master_discovery (fkie_master_discovery/maste
r_discovery)

ROS_MASTER_URI=http://192.168.50.202:11311

process[master_discovery-1]: started with pid [52
3629]
[WARN] [1681770378.690590]: Resolved host of ROS_
MASTER_URI 192.168.50.194=192.168.50.194 and orig
in discovered IP=192.168.50.202 are different. Fi
x your network settings and restart master_discov
ery!
```

/home/bartonlab-user/workspace/src/multimaster_fkie/fkie_ma

HSR Master - Master Discovery
Step 3

```
[WARN] [1681770407.228988]: 'NoneType' object has
                                                has
      bute '_md5sum'
      681770407.252189]: 'NoneType' object has
 no attribute '_md5sum'
[WARN] [1681770407.259253]: 'NoneType' object has
 no attribute '_md5sum'
[WARN] [1681770407.275076]: 'NoneType' object has
 no attribute '_md5sum'
[WARN] [1681770407.289598]: 'NoneType' object has
 no attribute '_md5sum'
[WARN] [1681770407.297388]: 'NoneType' object has
 no attribute '_md5sum'
[WARN] [1681770407.307266]: 'NoneType' object has
 no attribute '_md5sum'
[WARN] [1681770407.317986]: 'NoneType' object has
 no attribute '_md5sum'
[WARN] [1681770407.336817]: 'NoneType' object has
 no attribute '_md5sum'
[WARN] [1681770407.354506]: 'NoneType' object has
 no attribute '_md5sum'
[WARN] [1681770407.367252]: 'NoneType' object has
 no attribute '_md5sum'
[WARN] [1681770407.376373]: 'NoneType' object has
 no attribute '_md5sum'
```

bartonlab-user@me-bartonlab33: ~ 50x27

```
bartonlab-user@me-bartonlab33:~$
```

/home/bartonlab-user/workspace/src/multimaster_fkie/fkie_m

Laptop Master - Master Sync
Step 4

```
 * /master_sync/do_not_sync: []
      r_sync/ignore_publishers: []
      r_sync/ignore_services: []
 * /master_sync/ignore_subscribers: []
 * /master_sync/ignore_topics: []
 * /master_sync/ignore_type: []
 * /master_sync/resync_on_reconnect: True
 * /master_sync/resync_on_reconnect_timeout: 0
 * /master_sync/sync_hosts: []
 * /master_sync/sync_nodes: []
 * /master_sync/sync_remote_nodes: False
 * /master_sync/sync_services: []
 * /master_sync/sync_topics: []
 * /master_sync/sync_topics_on_demand: False
 * /rosdistro: noetic
 * /rosversion: 1.15.14

NODES
  /
    master_sync (fkie_master_sync/master_sync)

ROS_MASTER_URI=http://192.168.50.202:11311

process[master_sync-1]: started with pid [524388]
```

bartonlab-user@me-bartonlab33: ~ 49x27

Laptop Master - Master List
Step 5

```
/master_discovery/list_masters
      e: "192.168.50.202"
    uri: "http://192.168.50.202:11311/"
    last_change:
      secs: 1681770380
      nsecs: 770550251
    last_change_local:
      secs: 1681770365
      nsecs: 342280864
    online: True
    discoverer_name: "/master_discovery"
    monitoruri: "http://192.168.50.202:11611"
  -
    name: "192.168.50.194"
    uri: "http://192.168.50.194:11311/"
    last_change:
      secs: 1681770364
      nsecs: 701115131
    last_change_local:
      secs: 1681770364
      nsecs: 701115131
    online: True
    discoverer_name: "/master_discovery"
    monitoruri: "http://192.168.50.202:11612"
bartonlab-user@me-bartonlab33:~$
```

/home/bartonlab-user/workspace/src/khi_robot-master/khi_rob

HSR Master - Rostopic List
Sanity Check - Step 6

```
/tf2_web_republisher/result
                              n/obstacle_circle/
/tmc_map_merger/inputs/base_scan/obstacle_circle/
parameter_updates
/tmc_map_merger/inputs/base_scan/parameter_descri
ptions
/tmc_map_merger/inputs/base_scan/parameter_update
s
/tmc_map_merger/inputs/head_rgbd_sensor/obstacle_
circle/parameter_descriptions
/tmc_map_merger/inputs/head_rgbd_sensor/obstacle_
circle/parameter_updates
/tmc_map_merger/inputs/head_rgbd_sensor/parameter
_descriptions
/tmc_map_merger/inputs/head_rgbd_sensor/parameter
_updates
/tmc_remote_exec/run/cancel
/tmc_remote_exec/run/feedback
/tmc_remote_exec/run/goal
/tmc_remote_exec/run/result
/tmc_remote_exec/run/status
/update_map
/urg_cloud
<hsrb>~$
```

bartonlab-user@me-bartonlab33: ~ 50x27

Laptop Master - Rostopic List
Sanity Check - Step 7

```
bartonlab-user@me-bartonlab33:~$ rostopic list
/action_state
/ar_marker
/attached_object_information
/attached_object_request
/base_local_path
/base_path_plan/cancel
/base_path_plan/feedback
/base_path_plan/goal
/base_path_plan/result
/base_path_plan/status
/base_path_planner/inflated_static_obstacle_map
/base_velocity
/client_count
/collision_environment_map_array
/collision_environment_marker_array
/collision_map
/connected_clients
/current_floor
/current_room
/current_topological_position
/depthcloud_encoded
/depthcloud_encoded/compressed
```

Fig. 6. Steps to setup multimaster configuration

at different frequencies, image was being published at 2 hz and camera info was being published at 30 hz and so the april tag ros package could not detect the tag. The transform tree [1] as generated by ROS can be checked on this link.

D. *Multi-Master Configuration*

To make HSR not a master we tried few different methods. Configuring the shell script which is there in /etc/opt/tmc/robot, named as roshost.sh. This file is used to read the ip-address from if-config and assign it to the environment variable. So, we changed it to the laptop's ip-address in order to have laptop as the master, but this method was causing the ROS running on HSR to stop.

So, we implemented a different method in which we had to install ROS and multimaster_fkie package on ”hmi_user” in HSR and then link the workspace to the docker running ROS in the environment variables files which exists in /etc/opt/tmc/robot/docker.hsrb.user. But for this to run automaticaly on the start of the HSR robot, it has to run all the scripts on start and do master discovery using multimaster_fkie package, this was also not happening.

In order to make HSR work in multimaster [2] mode, we tried doing it in a different way, where in we run everything in the laptop terminal itself. But there are few terminals which are run in ”hsrb_mode” in order to run master discovery. So, in the multimaster_fkie, package, there is a master discovery launch file for laptop and the HSR. This file does the job of discovering all the ROS-Masters on that network. So, if we run the master discovery launch file in the terminal of laptop, it will discover the ROS-Master running on laptop and when we run the master discovery launch file in the terminal which is in ”hsrb_mode”, it discovers the ROS-Master running on hsrb. After that we run ”master_sync” launch file from multimaster_fkie package in the laptop terminal without ”hsrb_mode”, so that all the masters are synchronised together in the laptop, in order to make laptop the master controller.

Once this is done, when we run ”rosservice call /master_discovery/list_masters” in the laptop terminal without ”hsrb_mode”, we can see the synchronized list 6 of all the discovered masters running on the same network and all the topics are also available on the same terminal and so we can use the laptop as the master controller.

IV. Future Work

A. *Namespace conflicts of running multiple programs*

Even though we have solved the conflict of running DuAro bringup launch file when run with HSR, there are few other files like DuAro moveit which also needs to have correct namespace, otherwise which can cause errors while running the duaro_hsr_transform publisher file.

B. *Transform between Toyota HSR and DuAro*

We need to refine the transform between DuAro and HSR which is currently happening using the april tag. The problem currently being faced is that the transform is being automatically computed by the april_tag_ros script but the transform is not accurate, so we need to calculate the transfrom using the kinematics of links of both DuAro and HSR and using the transform published by april_tag ros and refine it.

C. Script to Run Multimaster-Mode using one command

Write a shell script to run multi-master rather than running it in multiple terminals.

D. Using the feedback of shared sensing

We can use the feedback of shared sensing which can be accessed on the main controller (laptop) to tune other processes happening on different robots or machines in different parts of the environment.

References

[1] Tully Foote. tf. URL http://wiki.ros.org/tf.

[2] Sergi Hernandez Juan. Multi-master ros systems, 2015. URL http://www.iri.upc.edu/files/scidoc/1607-Multi-master-ROS-systems.pdf.