SC627 Assignment 4 report

180100049 – Hemanth Dontamsetti

After implementing balancing algorithm I encountered following problems and observations

1) Need of Total turn to go back by few steps

If the bot next desired position is behind the bot then using velocity converter it had to rotate 180 deg and then move.

Solution : Modified Velocity_convert function. If desired position is behind the bot then

$$v = -\cos(ang_{err} + \pi)\, Distance_{desired\ point}; (if\ -\pi \leqslant ang_{err} < -\pi/2)$$

$$v = -\cos(ang_{err} - \pi)\, Distance_{desired\ point}; (if\ \pi/2 < ang_{err} \leqslant \pi)$$

$$\omega = median(-ANG_{MAX}, ang_{err}, ANG_{MAX}); V_{linear} = median(-V_{MAX}, V, V_{MAX})$$

$$Took\ care\ such\ that - \pi \leqslant ang_{err} \leqslant \pi$$

Now linear velocity is negative and angular velocity is in required direction if desired position is behind the bot

2) Giving False report of consensus at start of the program and process dies/completes

Few bot initial condition is such that it is exactly at mid_point of left and right bot and initial velocity conditions of all bots is zero. So this makes the bot “feel” that it has reached consensus (because I used midpoint and velocity conditions to check consensus).

Solution : Added a time condition such than it only checks consensus condition after 5 seconds of node starting. This gives enough time to start all bots and move with non-zero velocity until balancing condition is reached.

3) To avoid run-time errors

Included conditions so that consensus is checked after data from all topics (/odom, /left_odom, /right_odom) are received.

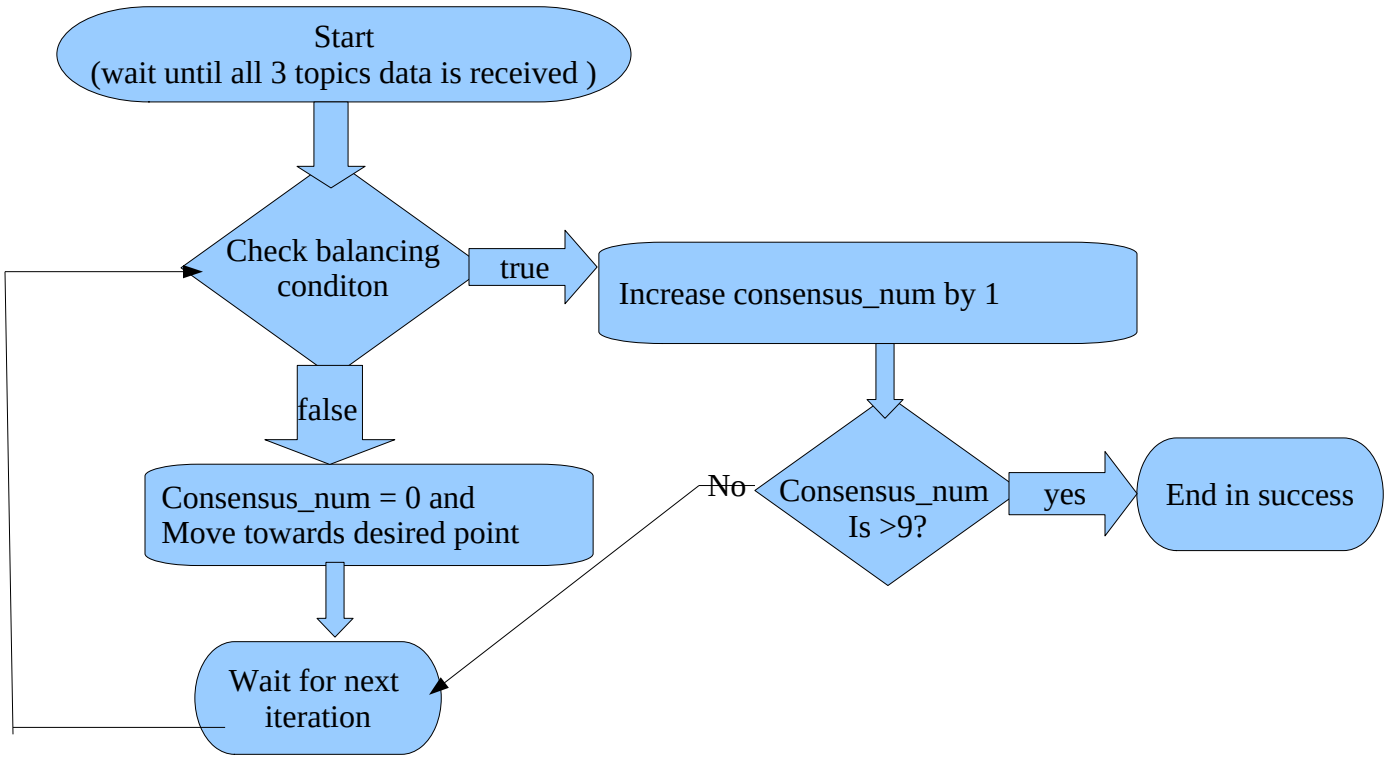
4) balancing condition for consensus

Desired point is mid point of left and right bots

$$Distance_{desired\ point} < 0.001 \wedge V_{leftBot} < 0.001 \wedge \omega_{leftBot} < 0.001 \wedge V_{rightBot} < 0.001 \wedge \omega_{rightBot} < 0.001$$

Success i.e. reached consensus only if this condition is **true 10 consecutive times**

➢ Flow chart of balancing algorithm:



➔ Global consensus took ~ 59 sec simulation time. Each turtlebot's simulation time to consensus are as follows

- Bot 2 – 56.2 s
- Bot 3 – 58.039s
- Bot 4 – 57.985 s
- Bot 5 – 58.051 s
- Bot 6 – 57.977 s
- Bot 7 – 57.151 s

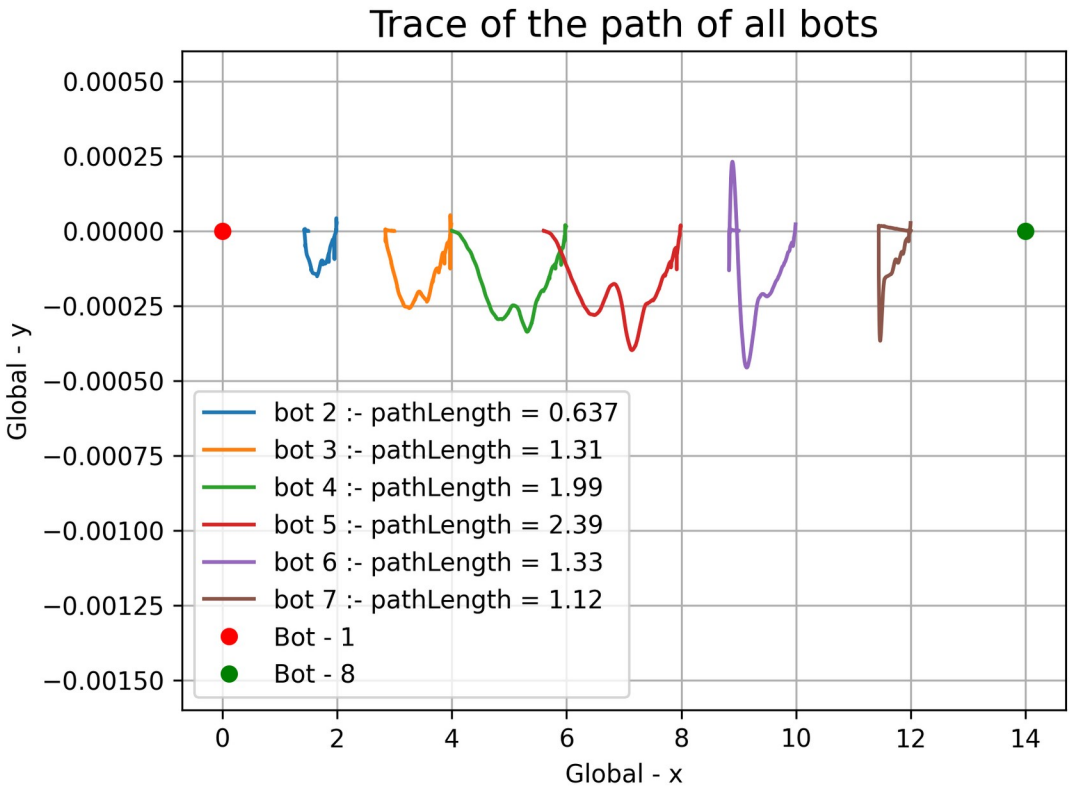


Fig 1: Total Trace all robots

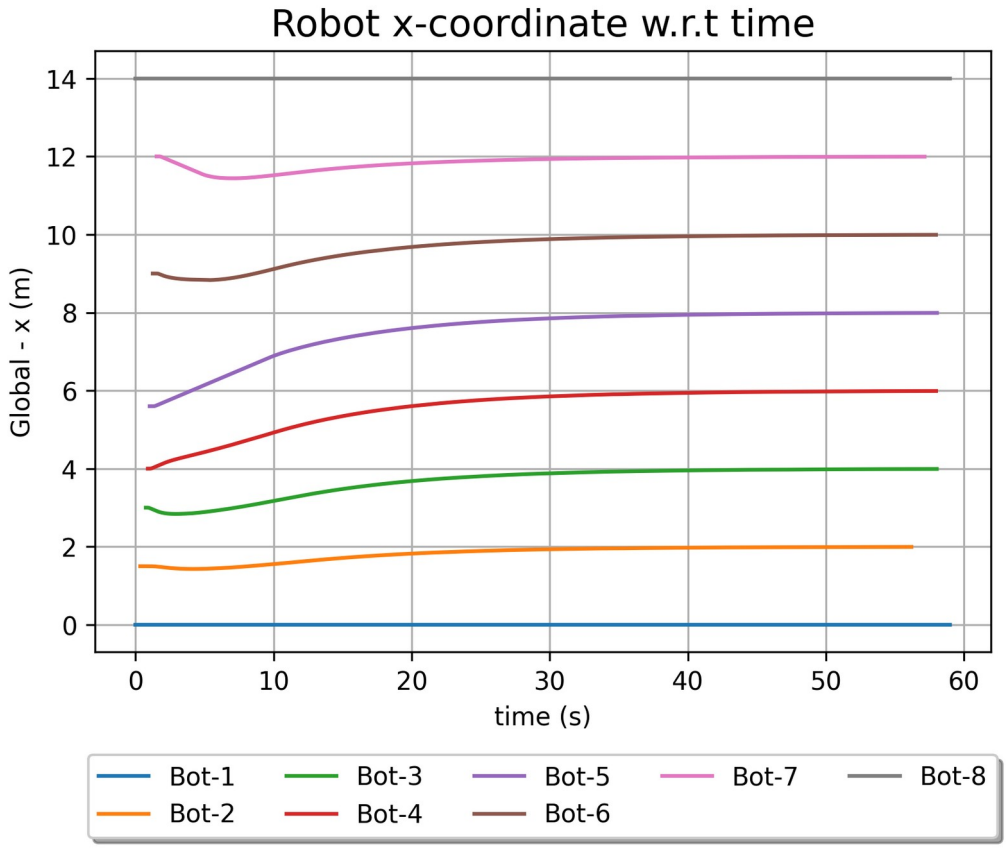


Fig 2: x v/s time of all bots