OBSTACLE AVOIDANCE:  
Gli eventi frontSonar permettono al robot di individuare gli ostacoli sul suo cammino. In presenza di uno di questi il robot cerca di capire se sia fisso o mobile.

Inizialmente assume che l’sotaclo sia fisse, aspetta dunque un piccolo intervallo di tempo per poi provare ad avanzare. Se non riceve altri eventi dal sona frontale, assume di essere riuscito ad evitare l’ostacolo.

**AvoidMobile**

Viceversa, se si ricevono altri eventi frontSonar, l’ostacolo viene considerato fisso (una parete ad esempio) ed il robot procede dunque a cercare di superarlo.

Per prima cosa tenta di aggirarlo da destra muovendosi parallelo all’ostacolo e cercando periodicaente la presenza di passaggi (es un a porta).

**AvoidFixRight**

Se durante queste manovre dovesse arrivare ad un vicolo ceco, oppure se non riesce ad individuare un passaggio entro un certo numero di tentativi, allora il robot ritorna indietro collocandosi nuovamente nella posizione da cui aveva cominciati le manovre.

**Failure + ResumeLastPosition**

In caso di fallimento delle manovre di aggiramento dell’ostacolo da destra, il robot non si perde d’animo e cerca di trovare un passaggio lungo il lato sinistro dell’ostacolo. Esegue le mosse in modo speculare rispetto a quanto visto.

**AvoidFixLeft**

In caso di un altro fallimento (vicolo ceco o tentativi massimi raggiunti), il robot interrompe la routine di pulizia come da requisito R-Obstacle.

**GivingUp**

In caso di esito nell’individuazione della porta, il robot avanza e poi percorre la stessa distanza parallela all’ostacolo ce aveva percorso per trovare il passaggio ma in senso opposto, in modo da tornare alla stessa altezza da cui era partito ed orientato nella stessa direzione di marcia lungo la quale aveva precedentemente rilevato l’ostacolo.

**GoToPrevLevelRigth/Left**

Per gestire il passaggio fra i sui stati interni, il robot utilizza dei messaggi privati swagmsg, mentre tutti i comandi generati dalla pressione dei bottoni sull’interfaccia grafica da parte dell’utente sono mappati in messaggi externalcmd.

Ricevere un qualsiasi externalcmd durante le operazioni di pulizia o di obstacleAvoidance, causa un’interruzione dell’operazione corrente ed una gestione del comando utente.

**whenMsg externalcmd -> receivedCmd**