Sliding Autonomy in Robot/Human Systems

Ian Stodart, Computer Science

Mentor: Dr. Daisy Tang

Kellogg Honors College Capstone Project

Problem

Autonomous robotic systems are being used more frequently in today's world, therefore, it is important for humans and robots to be able to work together to complete tasks. This project extends the prior work on sliding autonomy by extending the task allocation algorithm to involve both robots and human operators to efficiently solve the problem when necessary.

Approach

Task Allocation

Task allocation is the algorithm that creates teams of robots based on their distance from the box, and the strength of the robot. The algorithm determines the best team, the one closest to the box, and strength as small as possible to push the box. Then it sends a signal for the robots to push.

Desktop User Interface

Desktop User Interface allows the operator to assign tasks, run task allocation, and change the autonomy mode.

Mobile User Interface

Mobile User interface is used in Peer to Peer interactions and allows the operator to communicate with the robot in the field.

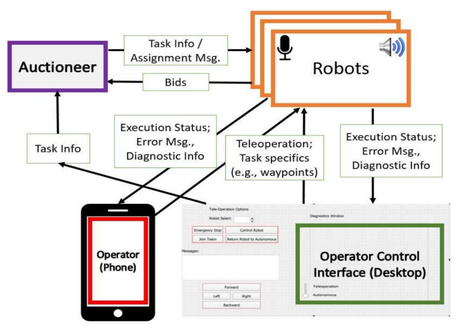
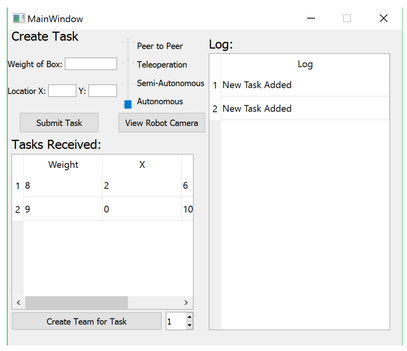


Fig. 1: The four components of our system with the flow of information.

Fig. 2: Desktop UI



Testing

To measure the performance of our sliding autonomy control, we collected data on the operator workload. Based on the amount of time that requires an active participation of the human operator, the operator workload is determined by having the operator fill out the NASA TLX (Task Load Index) Survey, with scores ranging between 1 and 100. The higher the number, the more demanding a task is to accomplish. The data in this area is very subjective, but still shows expected trends.

Figure 3

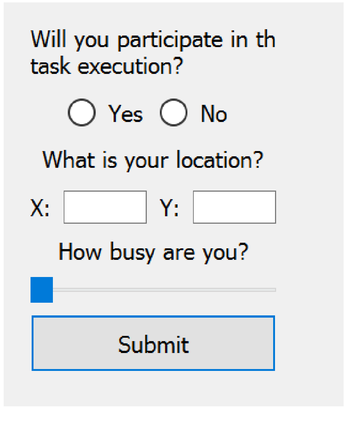


Figure 4

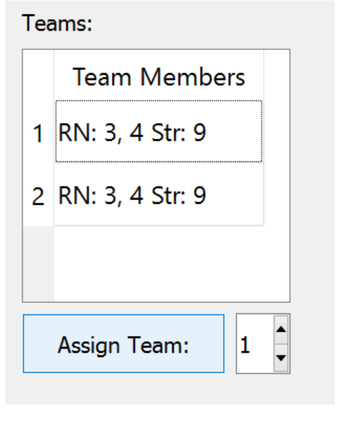


Figure 3: Operator Participation Window

Figure 4: Task Allocation Output window

Figure 5: Pairwise –
Each subscale is paired against another and the more relevant one is chosen

	Weight
Mental Demand	3
Physical Demand	2
Temporal Demand	4
Performance	5
Effort	1
Frustration	0

Figure 6: Rating Scale –
The rating is decided by how the user ranks each subscale on a scale.

	Rating	Weight	Adjusted
Mental Demand	55	3	165
Physical Demand	0	2	0
Temporal Demand	40	4	160
Performance	30	5	150
Effort	35	1	35
Frustration	0	0	0

Weighted Rating: 34.00

Analysis

The NASA TLX survey gives this project a 34 out of 100 as its rating. This means that the program demands a relatively low amount of effort from the user. This is due to the fact that the survey taker rated the frustration and physical demand as 0..

Discussion

The most contributing factor was the mental demand, likely due to the fact that the user has to enter tasks, choose which ones to assign, fill out a window about if he is going to participate, and then choose which team to assign. The temporal demand was the next highest because the operator has to quickly respond to the windows that pop up or the windows go away.

Conclusion and Future Work

The Task Allocation algorithm, and Desktop/Mobile UI's performed as expected, however when connecting them to another students work (to make the robots push the box), communication proved more difficult than expected and was not able to be done at this time. However, this project did accomplish the goal of creating the task allocation algorithm, and 2 UI's so that it creates tasks and assigns them to teams of robots and humans. In the future, I plan to fix the communication portion and get the UI and Task Allocation algorithm to assign teams, as well as full implementing voice recognition so that the human and robot can work together more naturally (like a group of humans) than communicating through a phone.

Acknowledgements

I would like to thank my mentor Dr. Daisy Tang, without her help and support this project would not have been possible. I would like to also thank Joseph Gunderson, he worked on the robots and without him we would never have been able to get as far as we did, and to all the other Computer Science department members who helped us debug in the last few weeks.

References

[1] Qt Project : IDE and framework for Desktop UI and Phone UI