34.4 Future work

This week I wrote a research plan. The main research work in the future will focus on the task allocation of heterogeneous robots and the task allocation of multi-robot with constraints. The research on constrained multi-robot task assignment also includes research on task release time in online task assignment.

In addition, I will also work on developing a multi-robot task assignment simulation environment to make the results of multi-robot task assignment more intuitive. After preliminary research, it is planned to use gazebo development under ros (melodic).

In addition, the program will continue to be optimized in the future to make it more modular, so that the simulation environment and algorithm can be separated.

35 Multi-robot task allocation with the concept of task execution

In the previous work, we did not consider the execution of the tasks after robot assigned the tasks. Although there is the concept of $TaskExecutionQueue$, it only represents the execution order after tasks are assigned to the robots. In other words, the tasks in $TaskExecutionQueue$ are not executed, but only stored in it.

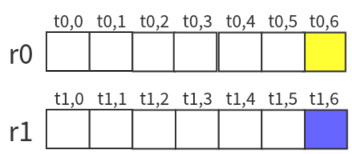

Fig. 76: TaskExecutionQueue

As shown in Fig.76 above, robot 0 and robot 1 have the $TaskExecutionQueue$ seperately. If we use robot self-coordination strategy, the new task which will be assigned next can place in any position in $TaskExecutionQueue$, if not, the new task will be place at the end of the $TaskExecutionQueue$ like the color squares in the figure above.

In such case, the key to allocating tasks depends on the utility of the new task itself, that is, the distance that the new task needs to be complete.

$$U_{ij} = \frac{1}{d_{ij}} \tag{42}$$

35.1 Introduce the concept of task execution

In practice, we will consider the following situation when we facing the task allocation: *we are always willing to assign tasks to robots with fewer unexecuted tasks.*

So, we introduce the concept of task execution that can represent the robots with dirrerent unexecuted tasks.

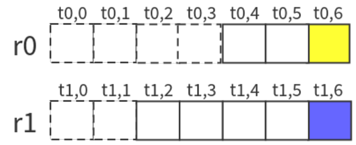

Fig. 77: TaskExecutionQueue with task execute

As shown in Fig.77 above, the dash line square represents the tasks which have been executed. For robot 0, there are 4 tasks $\{t_{0,0}, t_{0,1}, t_{0,2}, t_{0,3}\}$ have been executed. And robot 1 have 2 executed tasks $\{t_{1,0}, t_{1,1}\}$.

In this case, the key to allocating tasks can depend on the new task itself(the whole distance), and the unexecuted tasks in the $TaskExecutionQueue$.

There are two utility function can be chosen,

$$U_{ij} = \frac{1}{d_{ij}} + \frac{1}{d_{unexecuted}} \tag{43}$$

$$U_{ij} = \frac{1}{d_{ij} + d_{unexecuted}} \tag{44}$$

35.2 New objective function

Under the new utility, the original objective function of online task allocation need to change to adapt to the new allocation requirement.

The original objective function:

$$min \ \sum C_t \tag{45}$$

$\sum C_t$ represents the sum of the utility of all tasks assigned by the robot, which can be understood as the total execution distance of all robot $TaskExecutionQueues$.

The new objective function:

$$min \ C_{max} \tag{46}$$

C_{max} represents the distance of the robot with the longest total tasks execution distance.

35.3 Representation about task execution

It is assumed that there are two ways of representing task execution, one is that tasks are executed while being assigned, and task allocation takes time. The other is to assign tasks and execute them later. Tasks allocation do not take time.

We mainly consider the latter ways, that is, assign tasks and execute them later, becasue the time of task allocation often much shorter than the time for task execution.

Here we use Fig.78 to show the effect of task execution. There are 6 robots in a multi-robot system, represented as $\{r_0, r_1, r_2, r_3, r_4, r_5\}$. Circles represent the tasks,for example $t_{1,0}$ means the task with number 1 released from task point 1, and the different color represent the tasks released from defferent time. Squares represent a unit of time for task execution. Each task requires two units of time to complete the task, and each robots have a unit of time to execute task after allocation.

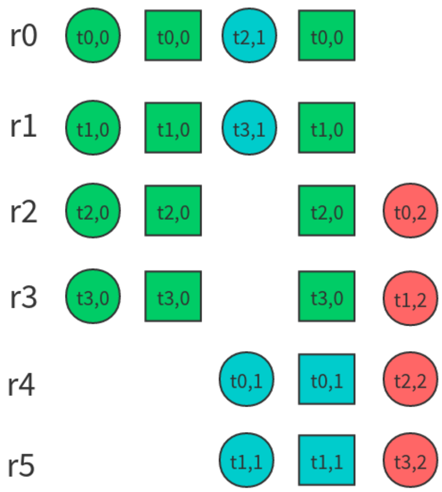

Fig. 78: Task allocation with task execution

In the above figure, tasks $\{t_{0,0}, t_{1,0}, t_{2,0}, t_{3,0}\}$ were firstly assigned to robot $\{r_0, r_1, r_2, r_3\}$ depends on utility. Then, these robots execute the assigned tasks in a unit of time. After that, the second set of tasks were released and assigned to $\{r_4, r_5, r_0, r_1\}$. Here, r_4 and r_5 are easier to get tasks because they have not been assigned tasks before. Then, all robots execute the assigned tasks in a unit of time again. Here we can see, $\{r_2, r_3, \}$ have executed all tasks they have, and $\{r_4, r_5, \}$ have executed half of tasks they have. But $\{r_0, r_1, \}$ still have a task to do seperately. So the third set of tasks were assigned to $\{r_2, r_3, r_4, r_5, \}$ depend on the new utility.

35.4 Comparison case

We give the comparison case that task allocation still use the original utility and do not introduce the task execution. The result is shown in the Fig.79 below,

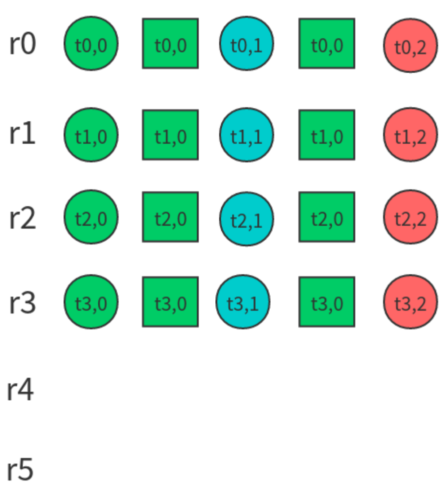

Fig. 79: Task allocation without task execution

The number of robots and the number of tasks are the same as before. In this case, $\{r_0, r_1, r_2, r_3\}$ are close to the three set of tasks and assigned them depend on the original utility. $\{r_2, r_3\}$ does not have any tasks.

We can see that, in this case, all robot just executed the first set of tasks before the third set of tasks assigned. When the third set of tasks is assigned, each robot still has two tasks to perform. However, in the case of task allocation with task execution, the longest execution time robot is $\{r_4, r_5, \}$ which just need to execute one and a half tasks seperately. Maybe the sum of task execution distance of comparison case is shorter than the task allocation with task execution case. But under the premise of the new objective function, the latter performs better.

35.5 Conclusion

The above discussion is a brief summary of the task execution concept currently being introduced. The two forms of the utility function are not discussed. At the same time, the representation of task execution gives certain assumptions, and no further content has been discussed. Next, on the basis of continuing to improve the problem assumption and model, I will open a new branch in the previous task allocation program to study this problem.

References

[1] D. P. Bertsekas, "The auction algorithm: A distributed relaxation method for the assignment problem," *Annals of Operations Research*, vol. 14, no. 1, pp. 105–123, Dec 1988. [Online]. Available: https://doi.org/10.1007/BF02186476

[2] ——, "The auction algorithm for assignment and other network flow problems: A tutorial," *Interfaces*, vol. 20, no. 4, pp. 133–149, 1990. [Online]. Available: http://www.jstor.org/stable/25061377

[3] M. M. Zavlanos, L. Spesivtsev, and G. J. Pappas, "A distributed auction algorithm for the assignment problem," in *2008 47th IEEE Conference on Decision and Control*, Dec 2008, pp. 1212–1217.

[4] Lingzhi Luo, N. Chakraborty, and K. Sycara, "Multi-robot assignment algorithm for tasks with set precedence constraints," in *2011 IEEE International Conference on Robotics and Automation*, May 2011, pp. 2526–2533.

[5] L. Luo, N. Chakraborty, and K. Sycara, "Provably-good distributed algorithm for constrained multi-robot task assignment for grouped tasks," *IEEE Transactions on Robotics*, vol. 31, no. 1, pp. 19–30, Feb 2015.

[6] D. P. Bertsekas and D. A. Castaon, "Parallel synchronous and asynchronous implementations of the auction algorithm," *Parallel Computing*, vol. 17, no. 6, pp. 707 – 732, 1991.

[7] S. Koenig, C. A. Tovey, M. G. Lagoudakis, E. Markakis, D. Kempe, P. Keskinocak, A. J. Kleywegt, A. Meyerson, and S. Jain, "The power of sequential single-item auctions for agent coordination," in *National Conference on Artificial Intelligence*, 2006.

[8] M. B. Dias, R. Zlot, N. Kalra, and A. Stentz, "Market-based multirobot coordination: A survey and analysis," *Proceedings of the IEEE*, vol. 94, no. 7, pp. 1257–1270, July 2006.

[9] B. P. Gerkey and M. J. Mataric, "A formal framework for the study of task allocation in multi-robot systems," *Intl.j.of Robotics Research*, vol. 23, no. 9, p. 2004, 2003.

[10] M. B. Dias and A. (tony Stentz, "A market approach to multirobot coordination," *Carnegie Mellon University*, 2001.

[11] Y. Sung, A. Budhiraja, R. Williams, and P. Tokekar, "Distributed assignment with limited communication for multi-robot multi-target tracking," 12 2018.

[12] R. Zlot, A. Stentz, C. Dias, M. Veloso, and T. Balch, "An auction-based approach to complex task allocation for multirobot teams thesis committee," Ph.D. dissertation, 12 2006.

[13] G. Korsah, A. Stentz, and M. Dias, "A comprehensive taxonomy for multi-robot task allocation," *International Journal of Robotics Research*, vol. 32, pp. 1495–1512, 10 2013.