MAS for Cooperative Planning

1. Short description

This system is written in Haskell and simulates a MultiAgent System which does distributed planning. I have **implemented case b** where each agent tries to auction as many of his tasks as possible. In fact, as it will be seen in the following section, all tasks are auctioned and as many tasks as possible are planned in a single planning round (cycle).

2. Basic Layout. CommunicationOverview. Plan strategies

There is one AF agent (Agent Facilitator). This is the only one who reads the system file. Thus, it is the only one who has knowledge of everything. However, he won't use any of this knowledge to the entire system's advantage: the planning will be done entirely by the AP agents which are created by the AF agent as it is commanded via the system file.

At each cycle the AF agent distributes all tasks from that cycle to the **first AP agent** he encounters. Having a clever layout is not worth the pain: since the AP agents will try to negotiate to the best of their abilities, there is no problem if all tasks are given to the first.

This is because at the start of each round the agents will call AP in order to find out which other APs are able to do each of the tasks the AP has. If there is no AP able to do a set of task then an initial planning is done trying to fit as many such tasks as possible (thus, taking them in the increased order of the capacity cost).

After the information about all other APs' capabilities is offered, each AP will send a call for proposal (using the Contract Net Protocol) to those agents able to do each task. Each proposal contains the task, the capability of the task and the id of the agent requiring the task to be done.

After all cfps are sent, each AP announces the AF that it has sent his proposals. In his turn, the AF announces all agents that the proposal phase is over and all of them start to plan the received tasks. Here, for a task we have three possible results: we can accept the task or we can deny it for two reasons: either it is not profitable to do this (the originator of the proposal can do the task at a lower cost) or there is no space for the task. Each AP replies this result to the originator then announces the AF that this subphase is over.

After the AF announces each AP that the proposal analysis phase is over, all APs select the lowest bid and announces the bidder that it can do the bid task. All other bidders are refused.

In the end, each AP tries again to plan the refused actions, as long as they are possible.

On the test files (found in the arhive and in the repo at https://github.com/mihaimaruseac/cdte), the profit obtained was always the highest possible.