32002: AI (Hierarchical task, multi-agent, and parallel Planning) **Spring 2014**

Lecture 38: April 19, 2014

Instructor: K.R. Chowdhary *: Professor of CS*

Note: *LaTeX template courtesy of IITJ, CSE dept.*

Disclaimer: *These notes have not been subjected to the usual scrutiny reserved for formal publications. They may be distributed outside this class only with the permission of the Instructor.*

38.1 Hierarchical Task Network Planning

In the hierarchical task planning, each level of hierarchy is decomposed into smaller levels. It is common for area like, military mission, administration, program development, where a task is reduced to small number of activities at the next level, so that the computational effort of arranging those activities is low. This results to reduction of complexity to linear time from exponential.

Consider an example of "building a house", where the task of house building can be decomposed to: acquiring land, preparation of design map, obtaining the NOC (no objection certificate) from municipal corporation, arranging house loan, hiring the builder, paying the builder, and so on (Figure 38.1). We understand that some of these activities can be done in parallel, but not all. Thus, there is a partial order relation between those actions. The decomposition can be expressed as $decompose(a, d)$, where action a is decomposed into a partial-order plan d. Various activities for building a house as per the plan in figure 38.1 can be formally described as follows.

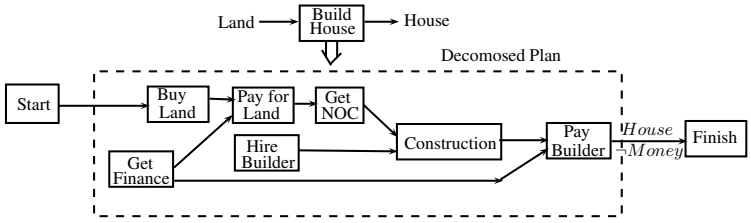

Figure 38.1: Decomposition of task for House Building.

$$Action(Buildhouse,\ Precond : Land,\\ Effect : House) \tag{38.1}$$

$$Action(Buyland,\ Precond : Money,\\ Effect : Land \wedge \neg Money) \tag{38.2}$$

$$Action(Getfiance,\ Precond : Goodcredit,\\ Effect : Money \wedge Mortgage) \tag{38.3}$$

$$Action(Hirebuilder,\ Precondition : Nil,\\ Effect : contract) \tag{38.4}$$

$$Action(Construction,\ Precond : NOC \wedge Builderhired,\\ Effect : Housebuilt \wedge \neg NOC) \tag{38.5}$$

$$Action(Paybuilder,\ Precond : Money \wedge Housebuilt,\\ Effect : \neg Money \wedge house \wedge \neg Contract) \tag{38.6}$$

$$Decompose(steps : \{A_1 : Get\ NOC,\ A_2 : Hirebuilder,\\ A_3 : construction, A_4 : Paybuilder\} \tag{38.7}$$

$$Orderings : \{start \prec A_1 \prec A_3 \prec A_4 \prec Finish;\\ Start \prec A_2 \prec A_3\} \tag{38.8}$$

Here, $A_i \prec A_j$ indicates that activity A_i precedes the activity A_j.

$$Links : \{start \xrightarrow{Land} A_1, Start \xrightarrow{Money} A_4,\\ S_1 \xrightarrow{NOC} S_3,\ S_2 \xrightarrow{Contract} S_3,\ S_3 \xrightarrow{Housebuilt} S_4,\\ S_4 \xrightarrow{House} Finish,\ S_4 \xrightarrow{\neg Money} Finish\} \tag{38.9}$$

38.2 Multi-Agent Planning Systems

There are a number of good reasons for having multiple agents creating plans:

1. The agents may represent real-life entities which mainly have their own interests at heart. Therefore, they appreciate maintaining their privacy and autonomy.
2. A distributed system may already exist, for which centralization would be too costly.

3. Creating and maintaining plans locally allows for a more efficient reaction in case of incidents, especially when communication is limited.
4. Finally, dividing the planning problem into smaller pieces and solving those in parallel may some times be more efficient, especially when the individual planning problems are loosely coupled.

In return for the lot many benefits of multi-agent systems, there are following difficulties in developing multi-agents planning:

1. How to put additional constraints upon the agents before planning, so that their resulting plans can easily be coordinated?
2. How to efficiently construct plans in a distributed fashion?
3. How to make collaborative decisions when there are multiple options for which each agent has its own preferences?

38.2.0.1 Multi-agent Planning

In general, a multi-agent planning problem can be defined as the problem of planning by and for a group of agents. Except for more centralized (multi-agent) planning problems, each agent in such a problem has in fact a private, individual planning problem. A typical individual planning problem of an agent includes a set of operations (with some costs attached, and a pre and post-condition) that it can perform, a set of goals (with reward values), and the current (initial) state of this agent.

Definition 38.1 *Multi-agent planning problem.*

Given a description of the *initial state*, a set of *global goals*, a set of two or more agents, and for each agent a set of its *capabilities* and its *private goals*, find a plan for each agent that achieves its private goals, such that these plans together are coordinated and the global goals are met as well.

□

The following statement perfectly captures the concept of multi-agent planning:

$$Multi\text{-}agent planning = planning + coordination \tag{38.10}$$

The solution to a multi-agent planning problem is a plan: a partially ordered sequence of actions that, when executed successfully, results in a set of achieved goals for some of the agents. Most techniques can deal with problems where the actions and goals of the agents are only weakly dependent upon each other, where the agents are cooperative, and where communication is reliable. However, in general a multi-agent planning approach may encounter a whole variety of situations along these three axes, with some characteristics as follows:

1. From independent to strongly related,

   Independent: no shared resources, no dependencies,

   Strongly related: joint actions, shared resources,

   E.g. lift a box together, car assembly,

2. From cooperative to self-interest agents,

3. In some settings the participating agents are only interested in optimizing their own utility, e.g., robots in the robo-cup versus companies in a supply chain management,

4. From no communication possible to reliable communication,

   In hostile environments agents may not or cannot communicate during execution. This may require all coordination to take place before the execution starts. E.g., robots rescuing people in disaster scenarios, or on a planetary exploration mission versus companies in a supply chain management.

38.2.1 Multi-agent Planning Techniques

Multi-agent planning techniques cover quite a range of solutions to different phases of the problem. In general, the following phases can be distinguished (generalizing the main steps) in task sharing:

1. Allocate goals to agents.

2. Refine goals into subtasks.

3. Schedule subtasks by adding resource allocation (possibly including the agents) and timing constraints.

4. Communicate planning choices (of prior steps) to recognize and resolve conflicts

5. Execute the plans.

Planning is a combination of phases 2 and 3 in the above, which are often interleaved. Any of these steps could be performed by one agent or some subset. Not all phases of this general multi-agent planning process need to be included. For example, if there are no common or global goals, there is no need for phase 1. Also, some approaches combine different phases. For example, agents can coordinate their plans while constructing their plans (combination of phase 2, 3, and 4), or postpone coordination until the execution phase (combination of phase 4 and 5), as, e.g., robots may do when they unexpectedly encounter each other while following their planned routes.

38.2.1.1 Goal and Task Allocation

Centralized methods often take care of the assignment of goals and tasks to agents during planning. There are, however, many other methods to assign tasks in a more distributed way, giving the agents a higher degree of autonomy and privacy. For example, complex task allocation protocols may be used, or auctions and market simulations.

An auction is a way to assign a task to the agent that attaches the highest value or lowest cost (called private value) to it. A *vickrey* auction is an example of an auction protocol that is quite often used in multi-agent systems. In a vickrey auction, each agent can make one closed bid, and the task is assigned to the highest bidder for the price of the second-highest bidder. This auction protocol has the good property that bidding agents should simply bid their true private values (i.e., exactly what they think it's worth to them), removing any need for additional reasoning about its worth to others.

References

[1] Chowdhary K.R. (2020) Logic and Reasoning Patterns. In: Fundamentals of Artificial Intelligence. Springer, New Delhi. https://doi.org/10.1007/978-81-322-3972-7_15