A Two-Level Negotiation Framework for Complex Negotiations[*]

Xiaoqin Zhang
Computer and Information Science Department
University of Massachusetts at Dartmouth
x2zhang@umassd.edu

Victor Lesser
Computer Science Department
University of Massachusetts at Amherst
lesser@cs.umass.edu

Tom Wagner
Automated Reasoning Group
Honeywell Laboratories
Tom.Wagner@honeywell.com

Abstract

In this paper, we present a negotiation framework in which the negotiation process is performed at two levels. The upper level deals with the formation of high-level goals and objectives for the agent, and the decision about whether or not to negotiate with other agents to achieve particular goals or bring about particular objectives. Negotiation at this upper level determines the rough scope of the commitment. The lower level deals with feasibility and implementation operations; negotiation at this level involves refinement of the rough commitments proposed at the upper level. The experimental work shows this two-level negotiation framework enables the agent to handle complicated negotiation issues and uncertainties in a more efficient way.

1 Introduction

Usually negotiation is structured as a single-level process. From the proposal to the final commitment, all related issues such as finishing time, achieved quality and offered price are determined in this process. Such negotiation can require a complicated reasoning process – particularly when the agent has multiple tasks, these tasks may be achieved in different ways, include sequencing constraints, and consume internal or external (shared) resources. Uncertainty in task execution may further complicate the negotiation process as behavior deviates from the expected. The deviation can cause re-negotiation over commitments or the adjustment of local activities so as to still meet the commitments. One major difference between this work and other work in negotiation is that in this work negotiation is not viewed as a stand-alone process. Rather, it is viewed as one of an agent's many interleaved activities – including scheduling, execution, and other negotiations. This view plus the complexities of negotiation mentioned earlier has led us to construct a two-level negotiation framework that makes the complexity inherent in this view more tractable. In this two-level negotiation framework, the negotiation process is performed at different abstraction levels to reduce the complexity of the search. An agent thus reasons about and negotiates over more important issues at the upper level, and then refines the rough commitments at the lower level in order to optimize its local plan and accommodate additional constraints and uncertainties. The focus of this work is on the decision-making process of negotiation, rather than the negotiation protocol or the language.

Let's look at an example to make these issues concrete. Agent A is Adam's personal assistant agent. Agent A is designed to carry out multiple tasks corresponding to Adam's multiple goals in his life. Adam is a professor of Asian culture and language and he also has a family. His department chair asks him whether he can deliver a college talk about his recent research activities, which requires some foreign material being translated. At the same time, he is planning to attend a research conference. Meanwhile, his wife discusses with him the arrangement for their son's birthday party. Thus, there are three candidate tasks that appear in the agenda of agent A: prepare a talk for Adam's college lecture, plan Adam's trip to a conference, and organize a birthday party for Adam's son. These tasks are associated with Adam's different roles and contribute to different goals. The contributions of these tasks are not interchangeable. Each task has a deadline request and has multiple alternative ways for it to be performed. Figure 1 shows these three tasks. The upper-level view describes the deadline for each task, the abstracted plans for each task, the duration of these plans and how they contribute to different goals. The lower-level view describes the detailed plan for each task and specifies the execution characteristics for primitive tasks.

In our example, agent A needs to make decisions about which tasks should be performed, when, and to decide how to perform them (which alternative to choose). The possible negotiations that agent A may be involved include:

1. Negotiation with the secretary agent about when the college talk should be delivered. This affects the deadline of the task *prepare talk*.

[*] This material is based upon work supported by the National Science Foundation under Grant No.IIS-9812755 and the Air Force Research Laboratory/IFTD and the Defense Advanced Research Projects Agency under Contract F30602-99-2-0525. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Defense Advanced Research Projects Agency, Air Force Research Laboratory/IFTD, National Science Foundation, or the U.S. Government.

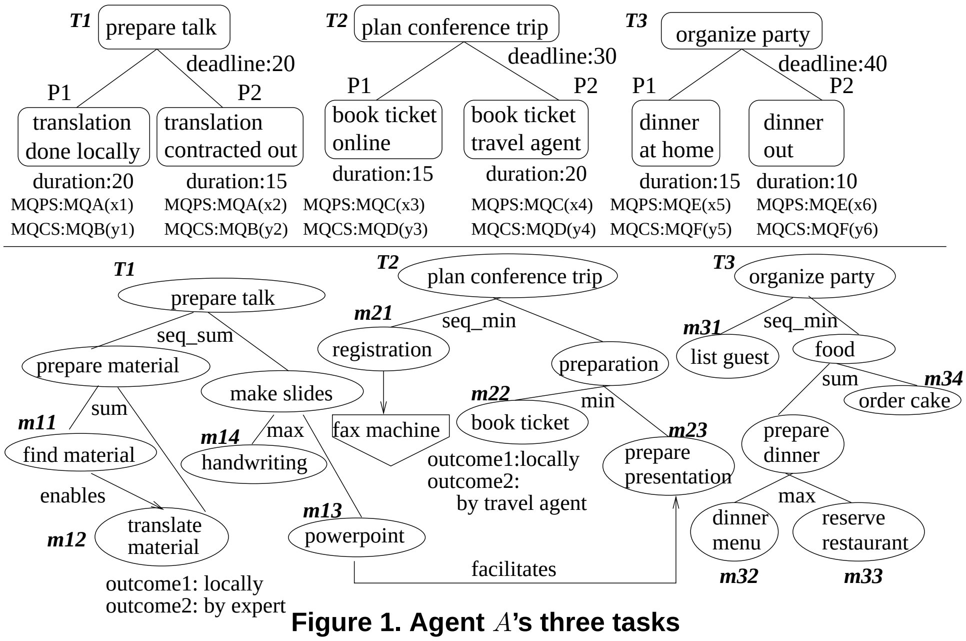

Figure 1. Agent A's three tasks

2. Negotiation with a translator agent about the task *translate material*, which includes when this task can be performed and how much it costs.
3. Negotiation with a travel agent about the task *book ticket*, which includes when this task can be performed and how much it costs.
4. Negotiation with agent W, the personal assistant agent of Adam's wife, about the task *organize party*, whether agent W can perform part of this task or the whole task.

In this paper, we take the position that it is reasonable to make high-level decisions about whether to (attempt to) perform tasks locally, or to negotiate over the tasks, without a detailed model of task attributes. All that needed is a rough view of the expected qualities of the different tasks, the expected qualities of alternative ways to perform the tasks, and any associated resource requirements. For example, agent A needs to perform task *prepare talk*, and there are two available plans for the *prepare talk* task:

1. P1: prepare the talk with the translation work done locally
2. P2: prepare the talk with the translation work contracted out to a translating agent

Each plan[1] has different quality, duration and cost characteristics. The plan $P2$ requests contracting a subtask *translate material* to another agent. From the high-level view, if agent A can find another agent to perform the subtask *translate material* before time 15 and with transferred utility less than 5, then plan $P2$ is the best choice. The availability of this commitment affects agent A's local plan. If such a commitment is not available, agent A needs to choose the other plan, $P1$, for task *prepare talk*. $P1$ takes longer to perform and hence makes it impossible for A to perform *plan conference trip* by its deadline. By comparing these two schedules – the one with the commitment on *translate material* to the other local schedule without the commitment on *translate material*, agent A can determine how important it is to obtain a commitment on *translate material* and perform $P2$ instead of $P1$.

However, not all issues can be modeled or totally decided on the upper level. There are two kinds of issues related to decision-making in negotiation: 1) Those issues, which have strong influence on local plan selection and involve utility transferred between agents (i.e. an important non-local task or an important resource that needs to be obtained from another agent), should be negotiated first at the upper level and rough commitments should be constructed for them. 2) However, we argue that those issues which have less influence on local plan selection and involve reasoning about the detailed structure of the lower-level activities, do not have to be directly reasoned about on the upper level and do not need to be decided on the upper level. These issues include:

1. *Internal relationships between subtasks that belong to different high-level tasks.* For instance, the subtask *PowerPoint* (make slides using PowerPoint) that belongs to *prepare talk* facilitates the subtask *prepare presentation* that belongs to *plan conference trip* because part of the slides for the lecture can be reused in the conference presentation if the slides are done in PowerPoint format. This relationship is not visible from the higher-level tasks.
2. *Uncertainty of the execution characteristics that are not visible on the higher level.* The agent is uncertain about the

[1] Planning from first principles is not addressed in this paper. The term "plan" here indicates a set of selected and ordered activities generated by the scheduler from a set of candidate TÆMS task structures – structures which identify the alternative ways that a task might be performed and their respective performance characteristics. The scheduler handles the choice and the sequencing of TÆMS tasks.

task's duration, cost and quality produced when it makes a plan about the task.

3. *Internal resource requirements associated with lower-level tasks.* For example, agent A needs to use the fax machine for task *registration* (Figure 1), but it shares the fax machine with several other agents. Given the knowledge of the general usage of fax machine, the agent knows it is unnecessary to reserve the fax machine when it builds its higher-level schedule.

Considering the above issues, the agent may need to revise its higher-level commitments through the lower-level negotiation. The agent may also have to reorder its lower-level activities, so as to optimize its local schedule and commitments, reduce failure possibilities, avoid conflicts and achieve higher utilities. A two-level negotiation framework is introduced in this paper. First we will present the supportive frameworks in Section 2, then we describe the basic underlying analytical ideas of the two-level negotiation framework in Section 3. Examples are used to explain how the framework operates in Section 4. Different reward models are discussed in Section 4.3.1. Section 5 shows how these different reward models affect the agent's performance and Section 6 summarizes this paper and discusses related work.

2 Supportive Frameworks

The multi-leveled negotiation is performed at different abstraction levels. In this work, the MQ framework [6] is used for the higher-level representation, while the TÆMS framework [1] is used to support the lower-level reasoning process. The proper integration of the MQ and TÆMS models and reasoning processes enables agents to reason about both organizational level task value and to handle detailed feasibility, analysis, and implementation of tasks. However, the basic approach is not restricted to these two frameworks, and we feel they can also be applied to other suitable task representation frameworks.

In the MQ framework, the execution of a task contributes, in a quantitative manner, to the achievement of one or more objectives of the agent. As part of this framework, there is a way of mapping this contribution to an overall utility increase associated with the potential execution of a task, given the agent's current state of achievement of different objectives. Each agent has a set of MQs or motivational quantities that it tracks and accumulates. MQ represents the progress toward a organizational goal and in certain cases may be used as a medium of exchange. MQs are produced and consumed by task performance where the consumption or production properties are dependent on the context. For each MQ$_i$ belonging to an agent, it has a preference function or utility curve, U_{f_i}, that describes its preference for a particular quantity of MQ$_i$. *MQ Tasks* are abstractions of a partial order set of primitive actions that the agent may carry out. MQ tasks may have *deadlines* and *earliest start times*. Each MQ task consists of one or more MQ alternatives, where each alternative corresponds to a different performance profile of the task. Each alternative requires some time or *duration* to execute, produces some quantity of one or more MQs, called the MQ *production set* ($MQPS$), and consumes some quantity of MQs, called the MQ *consumption set* ($MQCS$).

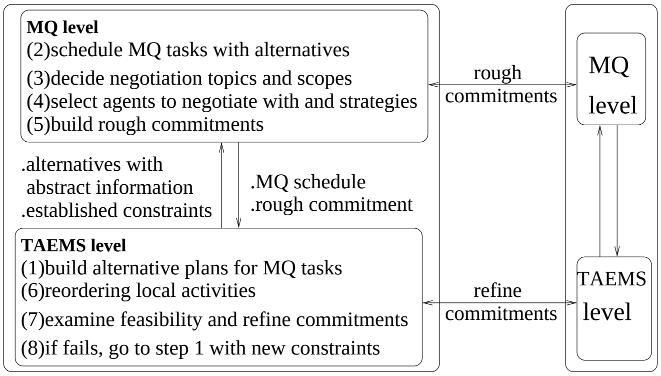
MQ level
(2)schedule MQ tasks with alternatives
(3)decide negotiation topics and scopes
(4)select agents to negotiate with and strategies
(5)build rough commitments
rough commitments
MQ level
.alternatives with abstract information
.established constraints
.MQ schedule
.rough commitment
TAEMS level
(1)build alternative plans for MQ tasks
(6)reordering local activities
(7)examine feasibility and refine commitments
(8)if fails, go to step 1 with new constraints
refine commitments
TAEMS level

Figure 2. A two-level negotiation framework

The TÆMS language [1] (See Figure 4) is a domain-independent task modeling language, which allows us to model agent's detailed activities. The agent's candidate tasks are described in hierarchical structures with alternative ways of accomplishing tasks. Three features are used to characterize the primitive tasks (methods): quality (q), duration (d) and cost (c) via discrete probability distributions. Quality describes the contribution of a particular method to the overall problem. Hard and soft interactions between tasks, called NLEs (non-local effects), are also represented in TÆMS and reasoned about during scheduling and negotiation. Hard task interactions delineate hard precedence constraints such as *enables* and *disables*. Soft task interactions denote situations where the result of one activity can *facilitate* or *hinder* another activity. Task resource consumption and production behaviors are modeled in TÆMS via *consumes* and *produces* task/resource NLEs.

3 Overview of Basic Ideas

In the two-level framework, an agent has an MQ level view of its local activities, which is a set of potential MQ tasks, each associated with certain $MQPS$ and $MQCS$. Figure 3 shows that agent A has three MQ tasks, $T1$, $T2$ and $T3$. $T1$ produces $MQ1$ from 6 units to 12 units, and it consumes $MQ2$ from 0 units to 6 units. The amount of the MQ varies depending on what plan is used to accomplish task $T1$. For each MQ task T, there is a TÆMS task structure that describes the detailed activities for this task, i.e. the task structure $TG1$ in Figure 4 describes the detailed activities in task $T1$. Different plans to accomplish the MQ task T can be generated from the TÆMS task group TG by the Design-To-Criteria (DTC) scheduler [5], and each plan

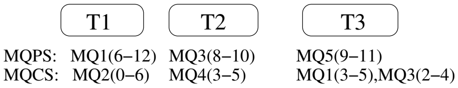

Figure 3. Example of MQ tasks

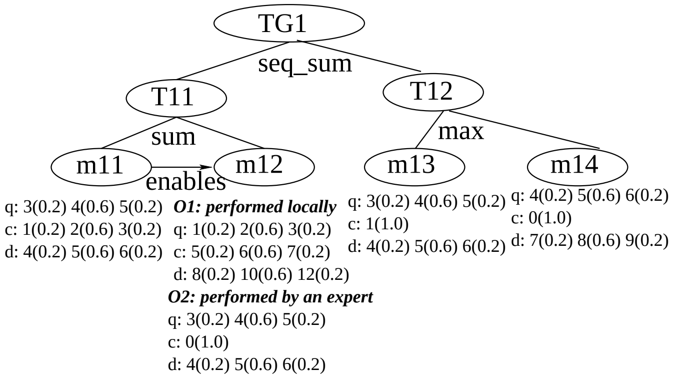

Figure 4. Task structure for T1

has different quality, duration and cost characteristics that affect the $MQPS$ and $MQCS$ of the task T. This is the first step [step 1] shown in Figure 2, which describes the two-level negotiation framework.

The extended MQ scheduler generates a partial order schedule that indicates what tasks the agent should attempt to execute, what plans are used to execute these tasks, and the order of the executions. This schedule represents the agent's best choice about what activities it should do to maximize its local utility increase [step 2]. Based on these schedules, the agent can reason about the utility of a specific commitment. Negotiation on the MQ level is a multi-dimensional negotiation that includes the amount of the transferred MQ, the temporal constraints and the quality constraints of the commitment [step3]. Also, the agent can select which agents to negotiate with and the appropriate negotiation strategy according to organizational relationships and the negotiation issues [step 4]. The MQ level negotiation builds rough (partial-specified) commitments for those issues that should or could be reasoned about the MQ level [step 5].

After building a local MQ schedule and rough commitments on the MQ level, the agent reorders its local activities on the TÆMS level [step 6]. In this reordering process, the agent optimizes its local schedule by taking advantage of the interrelationships among low-level tasks/methods. Also the agent verifies the feasibility of its local schedule given rough commitments from the MQ level and those additional constraints from the TÆMS level [step 7]. Negotiation on the TÆMS level involves refining those rough commitments as needed. If the agent can find a feasible local schedule by reordering and renegotiation on the TÆMS level, it can execute its local schedule and perform all of its commitments. If unexpected events cause conflict in the execution process, the agent needs to check if refining any commitments can solve the conflict. Otherwise, if the conflict can't be resolved given all current constraints, the agent needs to discard some commitments (decommits), establish other commitments on already scheduled local activities and go back to the MQ level to reschedule, and possibly result in constructing new commitments [step 8].

name	plan	q	c	d	MQPS	MQCS
					MQ1	MQ2
TG1_P1	(m11, m12, m13)	10	9	20	10	9
TG1_P2	(m11, m13)	8	3	10	8	3
TG1_P3	(m11, [m12]^2, m13)	12	3	15	10	9

Table 1. Alternative plans for task T1

4 Through the Process

In this section, we will discuss this two-level negotiation in greater detail using examples.

4.1 DTC Scheduler Builds Alternatives

The Design-To-Criteria (DTC) scheduler [5] is a domain-independent scheduler that aims to find a feasible schedule that matches the agent's particular criteria request. It is used off-line to build a library of alternative plans for achievement of a TÆMS task group. The three MQ level tasks $T1$, $T2$ and $T3$ are mapped into the task groups $TG1$, $TG2$ and $TG3$ in the TÆMS model. There is a subtask $m12$ of $TG1$ (See Figure 4) that potentially can be contracted to another agent who is an expert on task $m12$. The DTC scheduler works on $TG1$ according to the following different assumptions: $m12$ is executed locally, $m12$ is not executed, and $m12$ is contracted to another agent, and generates a set of alternative plans shown in Table 1. Each plan has different performance characteristics, corresponding to an MQ level alternative with different duration, $MQPS$, and $MQCS$. The quality and cost characteristics of a plan affect the $MQPS$ and $MQCS$ of the task, and the influence can be described using domain dependent functions.

4.2 MQ Level Scheduling

The MQ level scheduler does scheduling for these alternatives of $T1$, $T2$ and $T3$ to find the best schedule MQ_S1. Suppose the best schedule MQ_S1 includes the $TG1\_P3$ plan:

$TG1\_P3$[duration:15 earliest start time:0 deadline:20]
$TG2\_P2$[duration:10 earliest start time:0 deadline:30]
$TG3\_P1$[duration:15 earliest start time:10 deadline:40]

This is a partial order schedule. $TG1\_P3$ and $TG2\_P2$ need to be finished before $TG3\_P1$ starts. The reason is that $TG3\_P1$ consumes the MQs produced by $TG1\_P3$ and $TG2\_P2$. Agent A compares the utility of the best schedule including the contracting plan of $m12$ with the utility of the best schedule without the contracting plan of $m12$. The difference is the utility gained by contracting $m12$ to

another agent. It is used by the agents to guide the negotiation on the transferred MQ for contracting $m12$. The basic constraint of the quality request and the temporal constraint of $m12$ is established based on the TÆMS level schedule ($TG1\_P3$) and the MQ schedule (MQ_S1). Agent A posts this task allocation proposal as:
$m12, quality - request : 10, time - scope : [5, 15]$

4.3 MQ Level Negotiation

The negotiation on the MQ level includes the following concerns:

1. For each issue in negotiation, there are multiple features that could be negotiated about, such as the transferred MQ, which approach used to accomplish the task and the reward model. The negotiation is multi-dimensional.
2. For each negotiation session, there are different negotiation protocols available, such as single step negotiation or multi-step negotiation. The agent needs to find the appropriate negotiation protocol.
3. Although there is only one non-local task in negotiation in this example, it is often the case that there are multiple issues in negotiation and the negotiation on one issue affects the negotiations on other issues. The agent needs to decide the ordering of these negotiations and how it should negotiate on each issue.
4. Given that the other agents in negotiation may have different organizational relationships with this agent, the agent needs to choose appropriate negotiation attitudes toward other agents.

These issues have been studied as multi-dimensional negotiation, multi-step negotiation, multi-linked negotiation and integrative negotiation presented elsewhere. In this paper, we only describe how the agent selects an appropriate reward model that takes into account the possible further refinement of the rough commitment.

4.3.1 Reward Models Agents build rough commitments as a result of MQ level negotiation. We use the term "rough commitment" because the specifications can be ranges rather than points; these ranges allow further refinement. For example, a rough commitment c could specify the temporal constraint for the contracted task NL to be started and completed somewhere between $[t1, t2]$. If $f(c) > 0, t2 > t1 + d$ ($f(c)$ denotes the flexibility of c; d denotes the estimated duration of NL), it is possible to refine this commitment by restricting this range to $[t1 + x, t2 - y], (t2 - y) - (t1 + x) \geq d$; hence the flexibility of the commitment c (in terms of when NL can be performed) is reduced. Because the flexibility is related to the value/cost of the commitment, the agents need to come to an agreement on how the latter refinement is related to the value of the transferred MQ. There are two possible models:

1. *Pre-paid flexibility* model. The contractee agent E pays $v1$ of MQ_i for the contractor agent R to perform task NL during any time period (not shorter than d) within $[t1, t2]$ as agent E requests. This agreement provides agent E with the freedom to further refine this commitment, and agent R agrees to accommodate any request from agent E within the pre-defined range. No matter what request agent E will make, or even if agent E does not make any further requests, agent R will receive $v1$ of MQ_i as decided in the rough commitment.
2. *Dynamic flexibility* model. The contractee agent E pays $v2$ of MQ_i for the contractor agent R to perform task NL within the range of $[t1, t2]$. If agent E requests a restriction on this range to $[t1+x, t2-y], (t2-y)-(t1+x) \geq d$ and if agent R could accept this request, agent E will pay $((x + y) * \beta + 1) * v2$ of MQ_i to agent R. β is a parameter that can be adjusted, the agents can negotiate on the value of β. Agent R would decide to accept this additional refinement request or not, according to its current problem-solving context. If agent R does not accept this request, it is still obliged to perform NL during $[t1, t2]$ and in turn is guaranteed to get $v2$ of MQ_i as the rough commitment defines.

These two models provide different degrees of freedom for the agents. The agents can choose a model according to the constraints and uncertainties of their local activities during the negotiation process.

4.3.2 Reasoning about Uncertainty The uncertainty discussed here refers the uncertainty in the estimation of the execution characteristics (i.e. duration, quality, and cost) of an activity. In the lower-level reasoning process, uncertainties are represented as statistical distributions ($V : \{v_1(p_1); v_2(p_2); ...; v_n(p_n)\}$). Uncertainty information is abstracted as:

1. expected value: $E(V)$;
2. marginal value;
3. *probability of above expectation*: $-\sum_i p_i * log(p_i) * \frac{|v_i - E(V)|}{E(V)}$;
4. *measure of above uncertainty*: $\sum_{i|v_i>E(V)} p_i*(v_i - E(V))$.

This abstracted information is used in the upper-level reasoning process. Given the marginal value and the probability of the above expectation, the agent chooses the appropriate reward model. If the probability of the above expectation is large (bigger than a pre-set limit, this pre-set limit can adjusted by the agent based the learning from its experience.) or the measure of the above uncertainty is large, the agent chooses the pre-paid flexibility model because of the high probability of future change. Otherwise it chooses the dynamic flexibility model to save some cost on commitment. The marginal value is attached to the commitment to describe that a specified item in this commitment may need to be changed by the extent of the marginal value. If the contractee agent promises to accommodate this change when requested by the contractor agent (pre-paid flexibility model), it can charge a higher price for this commitment but it also needs to reserve enough room in its local schedule for the future change. Otherwise, the contractee agent can choose the dynamic flexibility model. In this way it does not promise to accommodate the future change. When the contractor agent requests a change, it checks its local schedule to see if this change can be guaranteed. If so, an extra cost is added when the change really happens.

4.4 TÆMS Level Negotiation

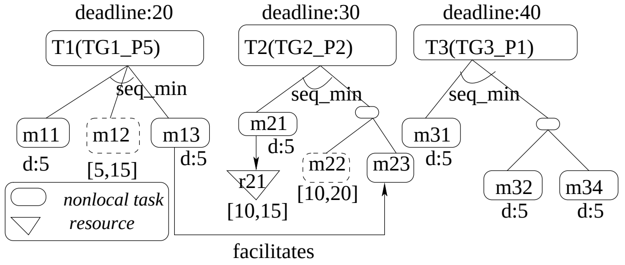

Figure 5. Agent A's TÆMS level tasks

Figure 5 shows agent A's current tasks and the required negotiation issues. Additional constraints come from the resource requirements and the relationships among those subtasks that belong to other high-level tasks: they are not visible to the MQ level scheduler so they are not reflected in the MQ level schedule. Two examples are shown in Figure 5: a facilitates relationship between $m13$ and $m23$, and resource $r21$ needed for the execution of method $m21$.

The reordering process considers all methods contained in the MQ level schedule. It takes into account the interrelationships among tasks, the resource request constraints and the rough commitments built at the MQ level negotiation. For example, resulting from the MQ level negotiation, agent B will perform task $m12$ for agent A between time 5 and 15, and agent C will perform task $m22$ for agent A between time 10 and 20. Given that the resource $r21$ is only available from time 10 to 15, agent A can't find a feasible local schedule. One solution is to negotiate with agent C to push the start time of $m22$ to 15 instead of 10 (suppose the duration of $m22$ for agent C is 5). If the commitment on $m22$ between agents A and C is the pre-paid flexibility model, then agent C would accept this request. Otherwise, if the commitment is associated with the dynamic flexibility model, agent C needs to reason about its local partial order schedule to determine if it can grant this request. If it can, agent C will get extra MQ from agent A as a result of the MQ level negotiation. If this refinement negotiation is successful, agent A can generate a new feasible local schedule:

m11[0-5] *m12[5-15]* m13[15-20]
m21[10-15] *m22[15-20]* m23[20-25]
m31[25-30] m32[30-35] m34[35-40]

Besides the additional constraints caused by resource requirements and the relationships among those subtasks that belong to different high-level tasks, the other reason for TÆMS level negotiation is the uncertainty of task execution.

5 Experimental work

The experimental work studies how the two-level negotiation mechanism affects the agent's performance compared to a one-level negotiation, and how the upper-level negotiation (the choice of reward model) affects the lower-level negotiation and hence affects the agent's performance. New tasks were randomly generated. Uncertainties are introduced by the execution component that generates the execution time for a task according to its statistical distribution. If a task takes longer than the expected time, it may cause other tasks to miss their deadlines. The lower-level negotiation occurs when this delay can be avoided by refining some rough commitments of non-local tasks. Four different policies are tested:

1. Fixed policy: The commitment built on the upper level (MQ level) is fixed; there is no lower-level re-negotiation to refine the commitment from the upper level.
2. Dynamic flexibility policy: The agent always chooses the *dynamic flexibility* reward model in the upper-level negotiation.
3. Pre-paid flexibility policy: The agent always chooses the *pre-paid flexibility* reward model in the upper-level negotiation.
4. Decision-making flexibility policy: In the upper-level negotiation, the agent chooses either the *dynamic flexibility* reward model or the *pre-paid flexibility* reward model according to the abstracted uncertainty information, as described in Section 4.3.2.

Figure 6 (each data point is an average over all cases with the same number of late tasks) shows that when the number of late tasks increases, the agent's performance decreases significantly without the lower-level negotiation (using the fixed policy). The reason is that the agent can not get the expected reward without finishing the task on time; additionally it has to pay the decommitment penalty. The lower-level negotiation helps the agent to adjust its previous commitment with the other agent, so as to avoid missing tasks' deadlines. As the number of late tasks increases, the performance of the dynamic flexibility policy decreases, because the dynamic flexibility policy can not guarantee the success of the lower-level negotiation. Whether the other agent accepts the adjust request depends on its current problem solving context. With the pre-paid flexibility policy, the agent's performance is almost stable regardless of the change of the number of late tasks. The agent always pre-pays for the flexibility to adjust the rough commitment whether it needs it or not. When the number of late tasks is small (less than 9), the agent actually wastes some of its potential gain by paying for flexibility it does not need. The decision-making flexibility policy brings the agent the nearly-best performance in all situations, because the agent can reason about when it may need flexibility and can pre-pay for it, or when it may not need extra flexibility and can save money on the contract.

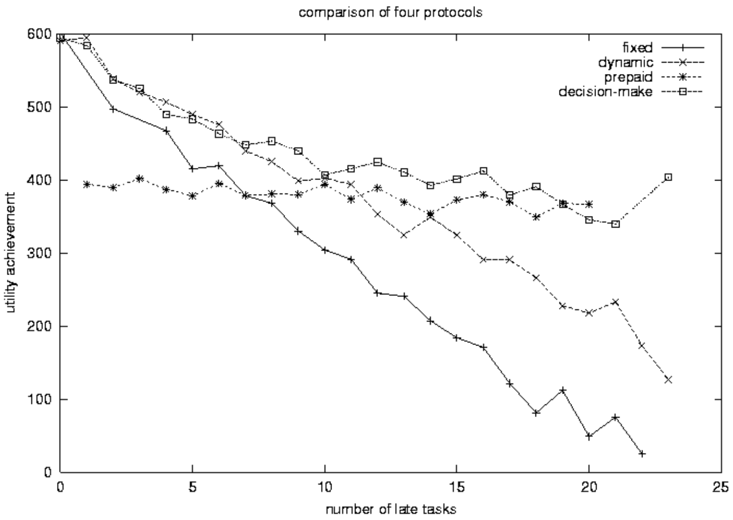

Figure 6. Agent's performance using different policies when uncertainty changes

6 Summary and Related Work

This paper explores a two-level negotiation approach. Other researchers have proposed multi-layered agent architectures, e.g. the InterRTaP [3] architecture includes three control layers. This architecture is based on BDI agent model, which is different from the utility-driven, quantitatively-reasoning agent control model in our work. Durfee and Montgomery[2] have presented a hierarchical protocol for coordinating multi-agent behaviors. DECAF [3] has also suggested a layered architecture based on separation along functional lines. However, these architectures have not addressed organizational concerns in the agent's goal selection process, as we do through the MQ framework, and none of them is focused on studying of the layered negotiation as our work does. The two-level scheduling process in this work is related to the early work in hierarchical planning[4] in the sense of decomposing problem into different abstraction level to reduce complexity. However, the contribution of this work is not on planning or scheduling technologies but on the integration of the negotiation decision-making process and the agent's layered scheduling process.

Rather than a stand-alone process, in this work, negotiation is viewed as one part of the agent's activity, which is tightly interleaved with the planning, scheduling and execution of the agent's activities, including other negotiations. This recognition has led us to a multi-level negotiation framework that allows us to handle the complexity inherent in this view. In this framework, an agent reasons about and negotiates over more important issues at the upper level (MQ level), and then refines the rough commitments at the lower level in order to optimize its local plan and accommodate additional constraints and uncertainties. Examples are used to explain how a number of different technologies, such as MQ, TÆMS and DTC can be incorporated to support sophisticated negotiation. Additionally, agents can choose an appropriate reward model in the higher-level negotiation according to the uncertainty measure; hence, the agent can pay for its local flexibility to accommodate the future uncertainty. The two-level negotiation framework enables the agent to reason about complicated negotiation issues and uncertainties in a more modular and computationally efficient manner. It also allows the agent to reason about the organizational concerns, implementation of objectives, and negotiation and re-negotiation decisions in an integrated way. This architecture opens up a wide variety of future work directions.

References

[1] K. S. Decker and V. R. Lesser. Quantitative modeling of complex environments. *International Journal of Intelligent Systems in Accounting, Finance, and Management*, 2(4):215–234, Dec. 1993.

[2] E. H. Durfee and T. A. Montgomery. A hierarchical protocol for coordinating multiagent behaviors. In *Proceedings of the 8th National Conference on Artificial Intelligence (AAAI-90)*, pages 86–93, Boston, MA, USA, – 1990. AAAI Press.

[3] K. Fischer, J. Muller, and M. Pisschel. Unifying control in a layered agent architecture. In *IJCAI95, Agent Theory, Architecture and Language Workshop 95*, 1995.

[4] E. Sacerdoti. The nonlinear nature of plans. In *Proceedings of the Fourth International Joint Conference on Artificial Intelligence*, 1975.

[5] T. Wagner, A. Garvey, and V. Lesser. Criteria-Directed Heuristic Task Scheduling. *International Journal of Approximate Reasoning, Special Issue on Scheduling*, 19(1-2):91–118, 1998.

[6] T. Wagner and V. Lesser. Evolving real-time local agent control for large-scale mas. In J. Meyer and M. Tambe, editors, *Intelligent Agents VIII (Proceedings of ATAL-01)*, Lecture Notes in Artificial Intelligence. Springer-Verlag, Berlin, 2002.