Advantages of Strategic Thinking in Multiagent Contracts (A Mechanism and Analysis)

Sunju Park, Edmund H. Durfee, and William P. Birmingham

Artificial Intelligence Laboratory
The University of Michigan
Ann Arbor, MI 48109-2110
{boxenju, durfee, wpb}@eecs.umich.edu

Abstract

Forming a contract among self-interested agents often requires complex, strategic thinking. For instance, a self-interested agent (contractee) will perform a task for another agent (contractor) only when doing it is in its own interest. Therefore, to form a successful contract, the contractor needs to provide incentives to the contractee, such as proposing a payment higher than the contractee's cost of doing the task.

In this paper, we focus on the contractor's strategic decision problem of what payment to offer to maximize its expected utility, and propose a method for a self-interested contractor to determine the best payment to offer. We present a four-step decision-making mechanism using a stochastic Markov-process (MP) model, which captures various factors that influence the utility value and uncertainties associated with them. The MP-based mechanism enables a contractor to choose different optimal payments depending on the payment(s) of the other contractor(s) or the contractees' costs of doing the task, and therefore to receive a better profit. Our experiments demonstrate that strategic thinking—thinking about not only oneself but also the contractees and the other competing contractors—is indeed advantageous.

1 Introduction

In multiagent systems, contracting enables agents to accomplish tasks they individually cannot do (Davis & Smith 1983; Sandholm & Lesser 1995a). Forming contracts among 'self-interested' agents, however, often requires more complex, strategic thinking than contracting among cooperative agents. For instance, a self-interested agent (contractee) will perform a task for another agent (contractor) only when doing it is in its own interest, and therefore, the contractor needs to provide incentives to the contractee, such as proposing a payment higher than the contractee's cost of doing the task.

As an example, consider a multiagent system consisting of self-interested agents who provide value-added information services (Atkins, *et al.* 1995). Assume there exists an agent (called a Task Planning Agent) who provides a list of possible sites in the Michigan area to observe a comet for a certain time period. The TPA will need information about observatories, their equipment, weather forecast, and so on, and it may want to monitor any changes in the weather forecast to provide a more accurate listing. Since monitoring is a specialty of another type of agent (called a Notification Agent), the TPA will buy the monitoring service from a NA.

Since the TPA is a self-interested agent who wants to maximize its utility (profit, in this example), it will try to charge more for its site listing, while paying as little as possible for the monitoring service from a NA. The problem is the TPA cannot do that without some strategic thinking about other competitors. If the TPA does not consider the other TPAs who provide the same listing service and need the same monitoring service, its profit will likely decrease, since no user will pay more money for the same list and no NA will do its monitoring service for less money. Moreover, the NA who is contracted to provide the monitoring service may retract from its contract later (while paying a retraction penalty) as more profitable contracts are announced by other TPAs. Therefore, the TPA needs to think about other self-interested agents (both NAs and other TPAs) and about the whole contracting process.

As shown in the above example, the multiagent contracting situations we are interested in have the following properties: contractors and contractees are self-interested; multiple contracts take place concurrently; and retraction from a contracted task may happen. Agents need to make strategic decisions in such contracting situations. A contractor needs to decide what payment to offer and to whom to make the offer, and a potential contractee needs to decide whether to accept the offer(s)[1]. In addition, once contracted, the contractee needs to decide which contract(s) to retract from, if any, to receive a more lucrative contract.

In this paper, we focus on the contractor's decision problem of what payment to offer to maximize its expected utility, and present a method for a self-interested contractor to determine the best payment. Since we believe strategic thinking—thinking about not only oneself but also the contractees and the other competing contractors—is desirable, we want to let the contractor use its knowledge about the potential contractees and the other contractors to find the best payment. To this end, we have developed a four-step decision-making mechanism using a stochastic Markov-process model. Furthermore, we have analyzed our method, comparing it with other methods. Our experiments demonstrate strategic thinking is indeed advantageous.

The rest of the paper consists of the following. Section 2 reviews some of the related work in multiagent contracts. Section 3 defines the contractor's decision problem and explains why such a decision is difficult. Section 4 describes our method based on Markov process model, and Section 5 examines some early experimental results. Section 6 discusses the future research issues and concludes the paper.

2 Related Work

This section reviews research on multiagent contracts, roughly dividing them into two: design of a global mechanism (i.e., protocol) and design of a local decision-making mechanism of each agent (i.e., strategy).

In systems consisting of self-interested agents (such as ours), agents can hardly assume what the other agents think and will do, except possibly their rationality. Therefore, a system architect needs to design a proper rule that ensures agents can concentrate on completing the tasks with some desired property (honesty, for example) rather than trying to manipulate the other agents or the system in an underhanded way. See (Rosenschein & Zlotkin 1994; Varian 1995) for global mechanism design for self-interested agents.

In terms of local decision-making, an agent with strategic thinking may get a higher payoff even in a task-oriented domain (Rosenschein & Zlotkin 1994) or even in a competitive market with large numbers of agents (Varian 1993). Therefore, an agent may want to think about what the others think about what it thinks about and so on (Gmytrasiewicz, Durfee & Wehe 1991). Or, it may model the contracting situation probabilistically, as done in decision analysis (Raiffa 1968).

The multiagent contract is in essence a game played among a set of contractors and a set of contractees, and such settings have been studied in noncooperative game theory and distributed AI (Rosenschein & Zlotkin 1994; Tirole 1988). In comparison, we model the contractor's decision problem stochastically rather than as a game (although the contractor does think about the contractees and the other competing contractors), since solving a game is hard and has additional difficulties such as equilibrium selection.

The contracting situation we are interested in consists of multiple, concurrent contracts, which demands a local decision-making mechanism that explicitly reasons about concurrent contracts and possible retractions, but there has been little research on such cases. Although Sandholm (Sandholm 1993) proposes a contracting strategy under multiple contracts, his agents do not capitalize on the opponents' costs or the impact of other contracts competing in the system. Recently, he proposes a leveled commitment protocol that allows agents to retract from a contract by paying a penalty (Sandholm & Lesser 1995b), but an agent's decision-making mechanism for such cases has not yet been developed.

3 The Contractor's Decision Problem

We assume the agents use a simple contracting protocol called Take-It-or-Leave-It (TILI). Under the TILI protocol, the contractor announces the task and its payment to potential contractees, and the potential contractees either accept or reject the offer (no counteroffers). Then, the contractor awards the task to one of the contractees who has accepted the offer. The contractee may want to retract from the contracted task while paying a retraction penalty. Retraction happens, for example, when the contractee finds a more lucrative task that cannot be done along with the contractor's task.

Consequently, the contractor's decision problem under the TILI protocol is to find the payment (ρ) that maximizes its expected utility (i.e., $maximize_\rho\ u(\rho)$).

In principle, the contractor's decision parameters can be diverse, such as the payment (ρ), the retraction penalty (δ), and the number of agents to announce its offer to (η)[2]. In this paper, however, we fix δ and η and assume that the contractor is only interested in its payoff. That is, the contractor's utility function is defined as

$$P_S(\rho) \times U(Payoff_S(\rho)) + P_F(\rho) \times U(Payoff_F(\rho)),$$

where $P_{S/F}(\rho)$ denote the probability of Success (S) and

[1] For now, we assume no counteroffer from the contractee side.

[2] Assuming the potential contractees are all homogeneous. If the contractor has information about individual agents, its decision parameter would be to whom (a subset of the potential contractees) to announce its offer.

Failure (F) of a contract, $Payoff_{S/F}(\rho)$ denote the payoff of S and F, given ρ.

The contractor's payoff for a successful contract ($Payoff_S$) is defined as its value of the task (V) minus the payment (ρ) to the contractee and minus the total overhead of the contracting process incurred by the contractor. At present, we are assuming that the only significant overhead cost is in communication such that the total communication costs (CC) are subtracted from the payoff. In addition, any retraction penalties (Δ) paid by the contractees are added. Note that retraction from the contractor side will not happen since its payoff will be the same no matter who it picks. The payoff of failure ($Payoff_F$) is minus the total communication costs plus any retraction penalties accrued, assuming the value of failure is 0. That is,

$$Payoff_S(\rho) = V - \rho - CC_S + \Delta_S$$

$$Payoff_F(\rho) = -CC_F + \Delta_F.$$

The contractor's decision—finding the payment with the highest utility—is complicated because many factors (in addition to the payment) influence the values of $P_{S/F}$ and $Payoff_{S/F}$ (and therefore the utility value). Figure 1 depicts those factors and their impacts on the utility value qualitatively. A black-colored link means a positive influence, while a gray-colored link means a negative one. For example, the payment has a negative influence on the payoff such that a higher payment results in a lower payoff.

As shown in Figure 1, the payoff will be higher if the contractor values the task higher, all else being equal. The probability of success would be higher if the potential contractees' costs of doing the task are lower. The total capability (TC) is the total cost that a contractee can take on per each time unit. Therefore, if TC is higher, retraction happens less often and thus the probability of success increases.

The factors in Figure 1 involve tradeoffs. For example, the contractor will prefer a higher retraction penalty for its task, since (1) once the task with a higher penalty is contracted, it is less likely to be retracted, and (2) even when retracted, the payoff (penalty collected by the contractor) will be higher. However, a higher retraction penalty may prevent the potential contractees from accepting its offer in the first place, which decreases the probability of success. Similarly, if the contractor offers a higher payment, it is more likely to be accepted (and therefore the probability of success would increase), but the payoff will decrease.

In addition, there are uncertainties associated with the factors. Some factors, such as the potential contractees' costs of doing the task, may be known to the contractor only probabilistically, and in such cases, the impact of the costs can be modeled only stochastically.

To find the best payment to offer, therefore, the contractor needs to model not only the payment but also other factors and their tradeoffs. In the following section, we present a method that captures those factors and their impacts on the utility value by stochastically modeling the future contracting process.

4 Mechanism: MP-based Modeling

We have developed a four-step mechanism for the contractor to compute the probabilities and payoffs, and therefore to find the best payment.

Step 1: Modeling the contracting process

The contractor can model various contracting processes using absorbing Markov chains with a set of

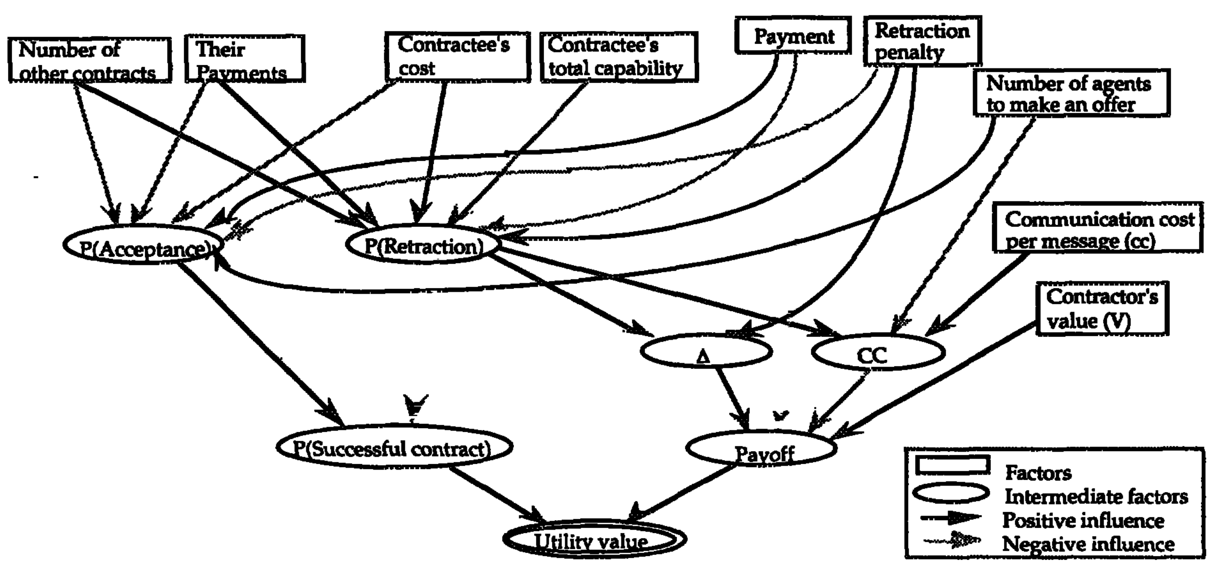


Figure 1: The factors that influence the utility value

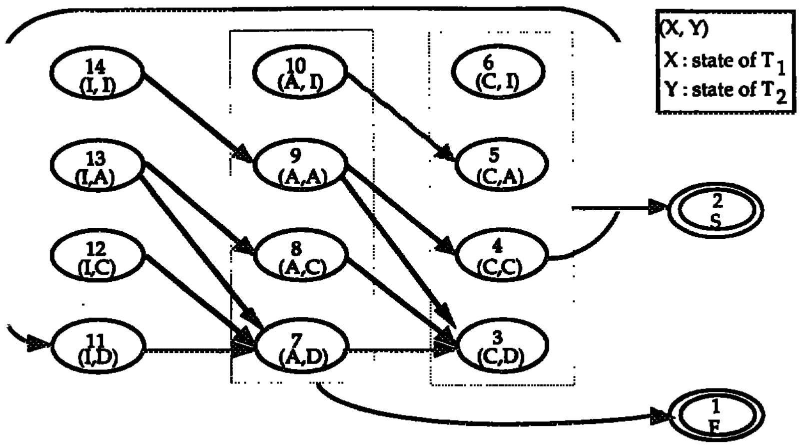


Figure 2: The MP model of the two-task contract with unit-time state transition

transient states and two absorbing states (*S* and *F*). An example of a contracting process model is shown in Figure 2.

Figure 2 depicts the case where there are two tasks in the system (T_1 is the contractor's task and T_2 is another contractor's task being contracted) and where every state transition takes one time unit. The *I, A, C, S, F* represent Initial, Announced, Contracted, Success, and Failure states, respectively. Note that we distinguish the state where a task is contracted (*C*) from the state where a task is successfully completed (*S*). The state *D* (done) is used to represent both *S* and *F* states of T_2, because the contractor of T_1 does not care about the result of T_2, and because it is generally a good idea to keep the number of states small.

When the contractor starts contracting T_1, it can be in any of the {14, 13, 12, 11} states; in general, the contractor can assign probabilities of being the initial state to those states. From the initial states, the process goes to the announced states, {10, 9, 8, 7}.

From {10, 9, 8, 7}, where T_1 is being announced, the process may go to state 1 (*F*) if no agent accepts the offer, or go to the contracted state {6, 5, 4, 3} as long as at least one agent accepts the offer. In this paper, we assume the contractor models the contractees as aggressive. An aggressive contractee is one who expects none of its bids will be awarded to itself (i.e., the probability of getting awarded is 0), and therefore accepts every contractor's offer as long as its cost for doing the task is less than the proposed payment[3]. So, if no agent accepts the offer, the contractor will not re-try the contract (since it knows there is no contractee who can do its task). That is, no transition from the announced states to the initial states (for re-contract) happens.

From the contracted states {6, 5, 4, 3}, the process goes to the success state (state 2) unless a retraction of T_1 happens at state 4. The contractor does not care about whether T_2 is retracted as long as T_1 is successfully finished.

Note that states 10, 6, and 5 will not be reached from the initial states; they are drawn for completeness. If a new task is introduced during the contracting process, however, those states can be reached.

Step 2: Computing the transition probabilities

The contractor defines the transition probabilities between the MP states by modeling the contractees' decisions at the announced state (whether to accept the offer) and at the contracted states (whether to retract the task(s)). The transition probability from state *i* to state *j*, P_{ij}, is computed from the factors identified in the previous section, such as the payment (ρ), the retraction penalty (δ), the contractees' costs of doing the task, their total capabilities (*TC*), the number of other tasks being contracted, and their payments.

As an example, let's compute the transition probability from state 13 to state 8, $P_{13,8}$. The transition happens when T_1 is announced from its initial state and T_2 is contracted from its announced state. That is,

$$P_{13,8} = P(announced_1) \times P(contracted_2),$$

where $P(announced_1)$ is 1 (since the transition from *I* to *A* always happens), and $P(contracted_2)$ is the probability of T_2 being contracted. Let $f_2^i(c)$ be the probability density function (PDF) of potential contractee *i*'s cost of doing T_2. Then, $P(contracted_2)$ can be computed as follows.

$P(A^i)$ = Probability agent *i* accepts the payment ρ of T_2 (i.e., probability agent *i*'s cost of doing T_2 is less than ρ)

$$= \int_{-\infty}^{\rho} f_2^i(c)dc .$$

$P(no\_contract_2)$ = Probability no agent accepts T_2

$$= (1-P(A^1)) \times (1-P(A^2)) \times \ldots \times (1-P(A^n))$$

$$= \Pi_i(1-P(A^i)) .$$

$P(contracted_2)$ = Probability at least one agent accepts T_2

$$= 1 - P(no\_contract_2)$$

When multiple tasks are being contracted to a single contractee, retraction (transition from the contracted

[3] As noted in (Sandholm 1993), the aggressive contractee is an approximation when there are many contractees.

state to the initial state) may happen. The transition from state 4 to 11 (i.e., retraction of T_1 when both tasks are contracted) happens when the following three conditions are met:

(1) Both tasks are awarded to the same contractee when they are contracted;

(2) The agent cannot perform both tasks (i.e., total capability is violated); and

(3) It decides to retract T_1 (i.e., the payoff of T_2 is higher).

Due to space limitations, we do not describe how the contractor computes the probabilities of (1), (2), and (3), and therefore the probability of retraction. The detailed explanation can be found in (Park, Durfee & Birmingham 1996).

Step 3: Computing the probabilities and payoffs of success and failure

Having the MP model and its transition probabilities, the contractor can compute the probabilities and payoffs of S and F.

Before continuing, let's define the transition probability matrix and the fundamental matrix.

The transition probability matrix, P, denotes the matrix of transition probabilities. Figure 3-(a) shows the canonical representation of a MP consisting of (s) transient states and (r) absorbing states: I is an $(r \times r)$ identity matrix (each absorbing state transitions to itself); O consists entirely of 0's (an absorbing state never transitions to a transient state); Q is an $(s \times s)$ submatrix which captures the transitions only among the transient states; and R is an $(s \times r)$ matrix which represents the transitions from transient to absorbing states. Figure 3-(b) shows the canonical representation of the transition probability matrix of Figure 2. We let T be the set of transient states and let T^C be the set of absorbing states (i.e., S and F).

$$\mathbf{P} = \begin{array}{c} \begin{array}{cc} (r) & (s) \end{array} \\ \left(\begin{array}{cc} \mathbf{I} & \mathbf{O} \\ \hline \mathbf{R} & \mathbf{Q} \end{array}\right) \end{array} \begin{array}{c} (r) \\ (s) \end{array}$$

(a) Canonical representation of P

$$\begin{array}{c|ccccc} & 1 & 2 & 3 & \cdots & 14 \\ \hline 1 & 1 & 0 & 0 & \cdots & 0 \\ 2 & 0 & 1 & 0 & \cdots & 0 \\ \hline 3 & 0 & 1 & & \cdots & \\ \vdots & & \cdots & & \cdots & \\ 14 & 0 & 0 & & \cdots & \end{array}$$

(b) Canonical form of Figure 2

Figure 3: The transition probability matrix

In Markov process theory, the matrix $(I - Q)^{-1}$ is called the fundamental matrix, M, and the (i,j)-th element of the fundamental matrix, μ_{ij}, means the average number of visits to transient state j starting from state i before the process enters *any* absorbing state (Bhat 1972). The fundamental matrix is very important, since it is used when computing the probabilities and payoffs of S and F as follows.

First, let f_{ij} be the probability that the process starting in transient state i ends up in absorbing state j. If the starting state is state 14 in Figure 2, for example, the probabilities of reaching S and F are $f_{14,2}$ and $f_{14,1}$, respectively.

Starting from state i, the process enters absorbing state j in one or more steps. If the transition happens on a single step, the probability f_{ij} is P_{ij}. Otherwise, the process may move either to another absorbing state (in which case it is impossible to reach j), or to a transient state k. In the latter case, we have f_{kj}. Hence,

$$f_{ij} = P_{ij} + \Sigma_{k \in T} P_{ik} f_{kj},$$

which can be written in matrix form as

$$F = R + QF,$$

and thus

$$F = (I - Q)^{-1} R = MR.$$

Therefore, the probabilities of S and F of a contract can be computed using the fundamental matrix (M) and the submatrix (R) of the original transition probability matrix.

Second, to compute the payoffs of S and F, we need $(-CC_S+\Delta_S)$ and $(-CC_F+\Delta_F)$. We first need to compute $\mu_{oi}^{(S)}$ and $P_{ij}^{(S)}$ ($\mu_{oi}^{(F)}$ and $P_{ij}^{(F)}$), where $\mu_{oi}^{(S)}$ is the number of visits to state i starting from the initial state, say 0, before the process enters S; and $P_{ij}^{(S)}$ is the conditional transition probability from i to j when the process ends up in S.

Those values can be computed by creating two new Markov chains ($P^{(S)}$ and $P^{(F)}$) from the original matrix P, each of which has one absorbing state, S and F, respectively. From $P^{(S)}$, we can obtain $P_{ij}^{(S)}$ and compute the new fundamental matrix $M^{(S)}$ (and therefore $\mu_{oi}^{(S)}$). $P_{ij}^{(F)}$ and $\mu_{oi}^{(F)}$ are computed similarly.

Now, let ω_{ij} of the reward matrix Ω be a reward associated with each transition $i \rightarrow j$, which can be either the minus communication cost $(-cc_{ij})$, or the minus communication cost plus the retraction penalty $(-cc_{ij} + \delta)$.

Then, $\sum_{j \in \{T, T^C\}} P_{ij}^{(S)} \cdot \omega_{ij}$ is the average reward of the one-step state transition from state i when the process ends up in S. Multiplying it by $\mu_{oi}^{(S)}$, we compute the one-step reward accrued from state i when the process ends up in S. Adding this value for every transient state i, we compute the total reward of S. That is, the total reward of S $(-CC_S + \Delta_S)$ can be computed as follows.

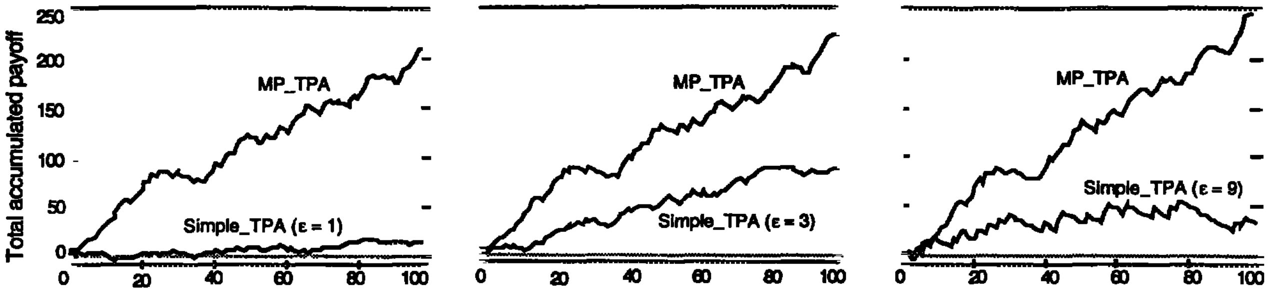
$$-CC_S + \Delta_S = \sum_{i \in T} \left(\mu_{oi}^{(S)} \cdot \sum_{j \in \{T, T^C\}} P_{ij}^{(S)} \cdot \omega_{ij} \right).$$

The reward of F $(-CC_F + \Delta_F)$ can be computed in a similar way.

$$-CC_F + \Delta_F = \sum_{i \in T} \left(\mu_{oi}^{(F)} \cdot \sum_{j \in \{T, T^C\}} P_{ij}^{(F)} \cdot \omega_{ij} \right).$$

Step 4: Finding the optimal payment

When the probabilities and payoffs of S and F are ready, finding the best payment is an optimization problem (Gill, Murray & Wright 1981). At present, we use a simple generate-and-test: we generate the utility values for various values of ρ, and choose the ρ with the highest expected utility. We are currently



(a) *Simple_TPA* with ε=1 (b) *Simple_TPA* with ε=3 (c) *Simple_TPA* with ε=9

Figure 4: Payoffs of the MP_TPA and the Simple_TPA

investigating an appropriate optimization technique for Step 4.

In this section, we have explained the method used by a contractor to determine the best payment to offer. The contractor uses its information about the contractees and the other contracts when constructing the MP model and its transition probabilities, and using the MP model, it computes the probabilities and payoffs of a contract, and thus the optimal payment.

The MP model is able to capture various factors that influence the utility value and uncertainties associated with them (in steps 1 & 2). In addition, we have developed a theoretically-sound method for computing the probabilities and payoffs from the MP model (in step 3).

5 Analysis: Advantages of MP Modeling

The MP-modeling method is based on the hypothesis that strategic thinking about the potential contractees and the other competing contractors is desirable. In our experiments, we demonstrate that this hypothesis is correct.

Let us return to the digital library's example contractors and contractees identified in Section 1, where a contractor might be a task-planning agent (*TPA*) that needs help monitoring weather changes, and a contractee could be a notification agent (*NA*) that can provide such a service. Our experimental setting has three potential contractees (NA^1, NA^2, NA^3). These contractees incur cost in providing a monitoring service, and are limited in the number of monitoring tasks they can each perform. This limitation is captured in a total capability (*TC*) constraint such that each *NA* cannot perform tasks whose total costs are higher than its total capability.

We assume that the contractees' actual costs are unknown to the contractors, but an informed contractor estimates the costs with a probability density function (PDF). For our experiments, the PDF of agent *i*'s cost for doing task *j*, $f^i(c_j)$, are given by

$$f^i(c_j) = \begin{cases} \frac{1}{6} & 5 \le c_j \le 11 \\ 0 & otherwise \end{cases}.$$

And the total capabilities (*TC*) of all three *NA*s are set to 12, which means a NA usually cannot perform two monitoring tasks at the same time.

With this setting in mind, we first establish that the MP-modeling method is better than a simple static strategy. To do this, we compare two contractors, the contractor with the MP-modeling method (called *MP_TPA*) and the contractor that does not think about the potential contractees nor the other contractor (*Simple_TPA*), each of whom has one task to be contracted (T_1 and T_2, respectively).

The *Simple_TPA* has a static decision-making mechanism: it offers to pay its value minus minimum communication cost minus some constant (i.e., $\rho = V - min\_CC - \varepsilon$). So, it will receive at most ε as its payoff when its task (T_2) is successfully completed, less than ε when retraction has happened during the contracting process[4], and a negative payoff when the contract fails. We call ε the maximum profit of the *Simple_TPA*.

To empirically compare the MP-based method with the simple strategy, we have randomly generated 100 sets of test inputs (for values and costs of T_1 and T_2), and run each with various *Simple_TPA*s with different ε. The values of T_1 and T_2 are randomly selected from between the contractees' lowest possible cost (5, in our setting) and some maximum value (20, in our setting).

As shown in Figure 4, the total payoff of the *MP_TPA* increases over time more rapidly than that of a *Simple_TPA*. When the maximum profit (ε) of a *Simple_TPA* is higher, the payoff of the *Simple_TPA* increases, but when ε becomes too high (9, for example), the payoff decreases. The decrease in the payoff comes from missed contracting opportunities because a *Simple_TPA* with a higher ε offers lower payment, and thus fewer contractees will accept the offer.

It is not surprising that *MP_TPA*s make better decisions than the *Simple_TPA*s, given that those decisions are more informed. Recall that there are two

[4] As defined in Section 3, when retraction happens, the contractor receives a retraction penalty but incurs additional communication cost. For our experiments, the penalty (+1) is set to be less than the additional communication cost (-2).

main considerations that go into the *MP_TPA*'s reasoning: considering what payments potential contractees might find acceptable (using the PDF of their costs), and considering how competition with other contractors might impact the outcome. A question arises as to whether both of these considerations are critical to the success of *MP_TPA*s.

To answer this question, our second experiments replace the *Simple_TPA* with an *Avg_TPA*, which uses the same PDF as the *MP_TPA* in considering the payments that potential contractees might accept, but does not consider the impact of the (in this case, one) other competing contract(s), and therefore ignores the possibility of retraction. The *Avg_TPA*, in other words, uses a Markov-process model for a single task.

Running 1000 runs to compare the total payoff of the *MP_TPA* with that of the *Avg_TPA*, using the same settings as before, we once again see that the *MP_TPA* achieves a higher total payoff, as shown in Figure 5. However, the difference in the profits of two contractors is much smaller than that of the *MP_TPA* and the *Simple_TPA*.

Of interest is that experiments with a relatively small test sample (e.g., 100 runs) sometimes result in a higher cumulative payoff for the *Avg_TPA*. We are investigating the conditions under which this happens, which may be used in developing an approximated MP-based mechanism.

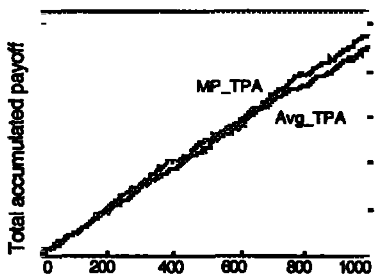


Figure 5: Payoffs of the MP_TPA and the Avg_TPA

Now that we have established the fact that MP-models capture both expectations about the contractees and about competing contractors, and that *MP_TPA*s thus achieve higher profit than our other sample contractors, another question arises. Did *MP_TPA*s perform better at the expense of their competition (i.e., at the expense of the less informed contractors) or at the expense of the contractees?

We use Figure 6 to form an initial answer to this question. Figure 6 shows the total payoff of the *MP_TPA*, the *Simple_TPA* (or the *Avg_TPA*), the sum of the payoffs of all the contractees (*NA*s), and the total payoff of the overall system for 100 runs each.

From Figure 6, we can make statements about the overall system performance. First, the overall system's payoff decreases with the *Simple_TPA* with higher maximum profit (ε) because of the missed contracting opportunities. Second, the overall system's payoff is higher when the contractors are more intelligent, because contracts are made more often and because when a contract is made, the *Avg_TPA* gets much higher payoff compared to the *Simple_TPA*s. That is, if we ignore the deliberation cost, having smart contractors increases the system's overall performance under the take-it-or-leave-it (TILI) protocol. In economical terms, smart contractors have extracted the surplus from the situation.

In terms of relative performance, we see that the increase in the payoff to the *Simple_TPA* or the *Avg_TPA*, if any, comes at the expense of the contractees (and not at the expense of the *MP_TPA*). And the payoff of the *MP_TPA* stays stable no matter which contractor it competes with. We account for this by the TILI protocol we are using, which leaves little choice to the contractees in improving their payoffs. As noted in (Varian 1995), the TILI protocol is better for maximizing the contractor's revenue when the margin between the values and the contractees' costs is higher. In our experiments, the contractees' costs are usually significantly lower than the value of the task (and the payment), and therefore using the TILI protocol enables the contractors gain profits at the expense of the contractees.

The above experiments demonstrate that the contractor who thinks about the other contractees and the competing contractors receives a higher payoff. In addition, we show that under the TILI protocol, the contractors get more profit at the expense of the contractees.

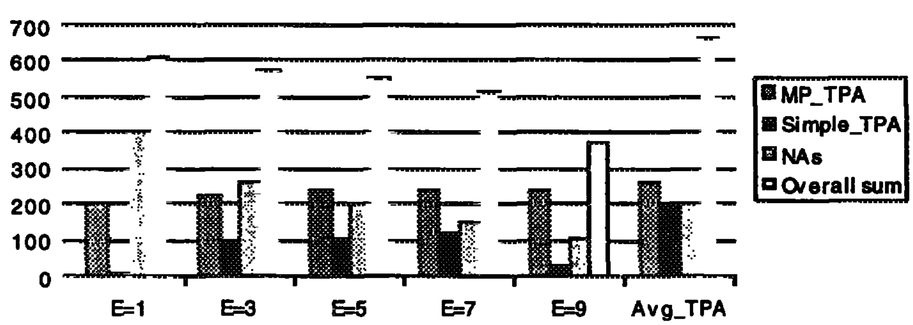


Figure 6: The payoffs of the MP_TPA, the Simple_TPA, and the NAs

So far, when comparing the MP-based method with others, we treat the deliberation cost as zero, which is not true in many cases. When deliberation takes time, the total payoff of the MP-based method will decrease. We are currently developing experiments where we can take deliberation cost into account. We expect the payoff difference to decrease as the deliberation cost increases, since the *Simple_TPA* (or the *Avg_TPA*) may be able to contract and finish more tasks while the *MP_TPA* is deliberating. Also of interest is the payoff per task; the payoff per task of the *MP_TPA* may remain higher than that of the *Simple_TPA*.

6 Discussion

In this paper, we have defined the contractor's decision problem, and proposed a method by which the contractor can find an optimal payment to offer. The contractor finds the best payment by strategically thinking about the contractees, the other competing contracts, and the contracting overhead (e.g., communication cost).

Our MP-based mechanism has introduced several new concepts and combined them with existing ones. For example, the MP-based model of the contracting process is new, while the interdependencies among multiple, concurrent contracts are re-defined using retraction and the total capability constraint of the contractees.

Our early results are promising. The MP-based mechanism can capture various factors that influence the utility value and uncertainties associated with them, and the contractor is able to choose different optimal payments depending on those factors. The experiments demonstrate that strategic thinking using MP modeling in general provides a better profit over a static decision or over a decision without modeling competing contractors.

At present, many research issues need further investigation. We are looking for an optimization technique for Step 4, and developing various MP models depending on the number of competing tasks or different amounts of information a contractor has. We are also investigating approximating some computations in the MP-based mechanism to minimize its deliberation cost.

Acknowledgments

This research has been funded in part by the joint NSF/ARPA/NASA Digital Libraries Initiative under CERA IRI-9411287.

References

Bhat, U. N. 1972. Elements of Applied Stochastic Processes: John Wiley & Sons Inc.

Atkins, D. E. 1996. Toward Inquiry-Based Education Through Interacting Software Agents. In IEEE Computer.

Davis, R., & Smith, R. 1983. Negotiation as a Metaphor for Distributed Problem Solving. Artificial Intelligence 20:63-109.

Gill, P. E., Murray, W., & Wright, M. H. 1981. Practical Optimization: Academic Press.

Gmytrasiewicz, P. J., Durfee, E. H., & Wehe, D. K. 1991. A Decision-Theoretic Approach to Coordinating Multiagent Interactions. In Proceedings of the IJCAI, 62-68.

Park, S., Durfee, E. H., & Birmingham, W. P. 1996. Use of Absorbing Markov Chains to Design a Multiagent Contracting Strategy, (In preparation). http://ai.eecs.umich.edu/people/boxenju/absorbing.ps

Raiffa, H. 1968. Decision Analysis: Random House.

Rosenschein, J. S., & Zlotkin, G. 1994. Rules of Encounter: Designing Conventions for Automated Negotiation among Computers: The MIT Press.

Sandholm, T. 1993. An Implementation of the Contract Net Protocol Based on Marginal Cost Calculations. In Proceedings of the AAAI, 252-262. Washington, D.C.

Sandholm, T., & Lesser, V. 1995a. Issues in Automated Negotiation and Electronic Commerce: Extending the Contract Net Framework. In Proceedings of ICMAS, 328-335.

Sandholm, T. W., & Lesser, V. R. 1995b. Advantages of a Leveled Commitment Contracting Protocol, (Technical Report 95-72), University of Massachusetts.

Tirole, J. 1988. The Theory of Industrial Organization. Cambridge, MA: The MIT Press.

Varian, H. R. 1993. Intermediate Microeconomics–A Modern Approach. New York, NY: W. W. Norton & Company, Inc.

Varian, H. R. 1995. Economic Mechanism Design for Computerized Agents. In Proceedings of the USENIX Workshop on Electronic Commerce. New York, NY.