Negotiation and Goal Relaxation

Gilad Zlotkin **Jeffrey S. Rosenschein**
Computer Science Department
Hebrew University
Givat Ram, Jerusalem, Israel
ph: 972-2-585-353
fax: 972-2-585-439
email: gilad@humus.huji.ac.il, jeff@humus.huji.ac.il

February 24, 2004

Abstract

Empty abstract.

In "Decentralized Artificial Intelligence II," Y. Demazeau and J. P. Muller, editors,
Elsevier Science Publishers B.V./North-Holland, 1991.

NEGOTIATION AND GOAL RELAXATION

Gilad ZLOTKIN and Jeffrey S. ROSENSCHEIN

Computer Science Department
Hebrew University
Givat Ram
91904 Jerusalem
ISRAEL

Previous work [15, 16, 18] discussed inter-agent negotiation protocols. One of the main assumptions there was that the agents' goals remain fixed—the agents cannot relax their initial goals. Goals can be achieved only as a whole and cannot be *partially* achieved.
Our purpose here is to give a general solution to the negotiation problem in non-cooperative domains where agents can relax their initial goals. An agreement may lead to a situation in which one or both goals are only partially achieved. We present a negotiation protocol that can be used in a general non-cooperative domain that includes goal relaxation. When goals are fixed, agents are negotiating only over a joint plan that achieves both goals. When goals can be relaxed, however, they are negotiating both over what *parts* of their goals will be satisfied, and (in parallel) over the joint plan that will be implemented to satisfy those parts.
Finally, we present several approaches to the goal relaxation problem and give some examples of domains in which each of the different approaches can be used.

1 Introduction

The subject of negotiation has been of continuing interest in the distributed artificial intelligence (DAI) community [13, 10, 7, 1, 6]. Some researchers, especially those concerned with "cooperative distributed problem solving," [2, 8] see negotiation as an important mechanism for assigning tasks to agents, for resource allocation, and even for deciding which problem-solving tasks to undertake. Much of this work relies as least partially on the assumption that the negotiation is taking place among agents that have been centrally designed to coexist in a single system; these agents are predisposed towards cooperative activity. At least implicitly, there is some notion of global utility that the system, through its design, is trying to maximize. Real conflict can exist, however, even among these agents, as a result of conflicting local goals.

Other researchers have focused on negotiation that might take place among agents that serve the interests of truly distinct parties [12, 14, 5, 15]. The agents are autonomous in the sense that they have their own utility functions, and no global notion of utility (not even an implicit one) plays a role in their design. Negotiation can be used to share the work associated with carrying out a joint plan (for the agents' mutual benefit), or to resolve outright conflict arising from limited resources (again, to the agents' mutual benefit). An autonomous agent in this context is assumed to be acting rationally only when it maximizes its own utility; in cooperative distributed problem solving, an agent might rationally make a decision that lowers its own utility for the global good.

Thus, despite the large amount of DAI research on this topic, there does not yet exist a universally accepted definition of what the word negotiation even means. As Gasser points out in [3], "negotiation [is] a term that has been used in literally dozens of different ways in the DAI literature." Nevertheless, it is clear to the DAI community as a whole that the operation of cooperating, intelligent autonomous agents would be greatly enhanced if they were able to communicate their respective desires and compromise to reach mutually beneficial agreements.

The work described in this article follows the general direction of [12, 15] in treating negotiation in the spirit of game theory, while altering game theory assumptions that are irrelevant to DAI.

Previous work [15, 16, 18] discussed inter-agent negotiation protocols. One of the main assumptions there was that the agents' goals remain fixed—the agents cannot relax their initial goals. Goals can be achieved only as a whole and cannot be *partially* achieved.

In order to cope with essential conflict between two agents' goals, we presented in [18] a negotiation protocol that allows the agents to reach an agreement in which they cooperate until a certain point, and then they flip a coin in order to decide which agent is going to continue the plan to achieve its goal. In conflict situations, only one agent will achieve its goal (with some probability).

Our purpose here is to give a general solution to the negotiation problem in non-cooperative domains where agents can relax their initial goals. An agreement may lead to a situation in which one or both goals are only partially achieved. We present here a negotiation protocol that can be used in a general non-cooperative domain that includes goal relaxation. Then we present several approaches to the goal relaxation problem and give some examples of domains in which each of the different approaches can be used.

2 Worth Function

Two autonomous agents A and B share the same world. The initial state of the world is s. Let ST stand for the set of all possible states of the world. Each agent has a single goal $g_i : i \in \{A, B\}$.

Definition 1

- *A goal is a closed formula in first-order logic (i.e., no free variables).*
- *A world state w will be said to* satisfy *the goal g if $w \models g$.*

- *The* worth *of the goal* g_i *to agent* i *is* $W_i \in \mathbb{R}$*, which is also the maximum expected cost that agent* i *is willing to pay in order to achieve his goal.*

Intuitively, each agent has only one goal that totally describes what he wants to achieve (of course, this goal may be a conjunction of subgoals).
The ability of an agent to relax his goal presupposes that there exist world states, other than those states that satisfy his goal, that are worth something to him. Thus, fundamental to the idea of goal relaxation is the existence of a Worth function over possible world states:

$$\text{Worth}_i\colon \text{ST} \rightarrow \mathbb{R}.$$

We will assume that the Worth function is defined for all states.
If a state $f \in \text{ST}$ satisfies g_i (i.e., $f \models g_i$) then $\text{Worth}_i(f) = W_i$, otherwise $\text{Worth}_i(f) < W_i$. $\text{Worth}_i(f)$ is an indication of what "part" of the goal g_i has been achieved, or how "close" state f is to the achievement of the whole goal. If $0 < \text{Worth}(f) < W_i$, we will say that the state f *partially* achieves g_i.
Our assumption from previous work, that *no* goal relaxation could occur, was actually a subcase of our current approach—there, the Worth function was a two-valued function, defined as follows:

$$\text{Worth}_i(f) = \begin{cases} W_i & \text{if } f \models g_i \\ 0 & \text{otherwise.} \end{cases}$$

This means that there does not exist a state that partially satisfies g_i. In the general case, Worth_i can be any real function that is bounded by W_i.[1]

3 General Definitions

Both agents have the same set of operations OP that they can perform. Operations o in OP move the world from one state to another; they can be concatenated into *plans*.

Definition 2 Plans

- *A* one-agent plan *to move the world from state* s *to state* f *in* ST *is a list* $[o_1, o_2, \ldots, o_n]$ *of operations from* OP *such that* $f = o_n(o_{n-1}(\ldots o_1(s)\ldots))$.
- *A* joint plan *to move the world from state* s *to state* f *in* ST *is a pair of one-agent plans* (P_A, P_B) *and a schedule.*

A schedule is a partial order over the union of actions in the two one-agent plans. It specifies that some actions cannot be taken until other actions are completed. Because it is a *partial* order, it of course allows simultaneous actions by different agents. If the initial state of the world is s and each agent i executes plan P_i according to the schedule, then the final state of the world will be f. We will sometimes write J to stand for a joint plan (P_A, P_B). $J(s)$ will stand for the final state f.

[1] There may be situations where no world state fully satisfies g_i—the goal can be only partially achieved. In this case $\forall f \in ST\colon \text{Worth}_i(f) < W_i$.

Definition 3 Costs

- *There exists a* cost function *over plans.*
- *For each one-agent plan* $P = [o_1, o_2, \ldots, o_n]$*,* $\mathrm{Cost}(P) = \sum_{k=1}^{n} \mathrm{Cost}([o_k])$*.*
- *For each joint plan* $J = (P_A, P_B)$*,* $\mathrm{Cost}_i(J)$ *is defined to be* $\mathrm{Cost}(P_i)$*.*

The above definition of the cost of a plan is not critical to the subsequent discussion (for example, whether or not the cost of a plan depends on the plan's initial state). The cost function may, in fact, have parameters other than the plan itself, such as the initial state, day of the week, and other domain dependent variables. What is important is the ability of an agent to measure the cost of a one-agent plan, and his ability to measure the cost of one agent's part of a joint multi-agent plan.

Definition 4 $s \rightarrow f$ *is the minimal* Cost *one-agent plan that moves the world from state* s *to state* f*.*

- *If a plan like this does not exist then* $s \rightarrow f$ *will stand for some constant plan* $\bowtie$ *which costs infinity.*
- *If* $s = f$ *then* $s \rightarrow f$ *will stand for the empty plan* Λ *which costs* 0*.*

4 Underlying Assumptions

In [15], we introduced several assumptions, some of which are in force for our discussion here as well (the final assumption was implicit in previous work):[2]

1. **Utility Maximizer:** *Each agent wants to maximize his expected utility.*
2. **Complete Knowledge:** *Each agent knows all relevant information.*
3. **No History:** *There is no consideration given by the agents to the past or future; each negotiation stands alone.*
4. **Bilateral Negotiation:** *In a multi-agent encounter, negotiation is done between a pair of agents at a time.*

5 One Agent Best Plan

If agent i is alone in the world, and only needs to achieve his own goal, he would bring the world to a final state f that maximizes his utility. Utility of an agent in general is the difference between the worth of a final state and the cost that an agent has to pay in order to bring the world to this final state.

[2] The relaxation of the Complete Knowledge assumption, in specific circumstances, was treated in [15, 17]. Future work will further examine the consequences of removing one or more of these assumptions, such as the No History assumption and the Bilateral Negotiation assumption.

If agent i were alone in the world, he would bring the world to a state f_i that satisfies the following condition:

$$\text{Worth}_i(f_i) - \text{Cost}(s \to f_i) = \max_{f \in ST} (\text{Worth}_i(f) - \text{Cost}(s \to f)) \, .$$

We use U_i to stand for $\text{Worth}_i(f_i) - \text{Cost}(s \to f_i)$, which is the maximum utility that agent i could achieve if he were alone in the world and tried to achieve only his own goal.

6 Negotiation over Relaxed Goals

Definition 5 *If there does not exist a world state that satisfies (partially or fully) both agents' goals, i.e.,* $\neg\exists f(\forall i \ \text{Worth}_i(f) > 0)$*, then we will call this a* conflict situation*.*

In non-conflict situations, the agent may negotiate over joint plans J that move the world from state s to state $J(s)$. To overcome the problem of indivisible operations, they may negotiate over *mixed joint plans.*

Definition 6 Deals

- *A* mixed joint plan *is written as* $(P_A, P_B){:}\,p$ *where* (P_A, P_B) *is a joint plan, and* $0 \leq p \leq 1 \ \in \mathbb{R}$*. Its semantics is that the agents will perform the joint plan* (P_A, P_B) *with probability* p*, or the symmetric joint plan* (P_B, P_A) *with probability* $1 - p$*.*
- *If* $\delta = (J{:}\,p)$ *is a mixed joint plan, then* $\text{Cost}_i(\delta)$ *is defined to be* $p\text{Cost}_i(J) + (1 - p)\text{Cost}_j(J)$ *(j stands for i's opponent).* $\delta(s)$ *will stand for* $J(s)$*.*

The agents are going to negotiate over two separate items:

1. What will be the final state of the world?
2. How will they share the work of bringing the world to this final state?

The utility for an agent of a mixed joint plan is simply the difference between the final state's worth to him, and the expected work to which the deal commits him.

Definition 7

- *If* δ *is a mixed joint plan, then* $\text{Utility}_i(\delta)$ *is defined to be* $\text{Worth}_i(\delta(s)) - \text{Cost}_i(\delta)$*.*
- *A mixed joint plan* δ *is* individual rational *if, for all* i*,* $\text{Utility}_i(\delta) \geq 0$*.*
- *A mixed joint plan* δ *is* pareto optimal *if there does not exist another mixed joint plan that dominates it—there does not exist another mixed joint plan that is better for one of the agents and not worse for the other.*
- *The* negotiation set NS *is the set of all the mixed joint plans that are both individual rational and pareto optimal.*

The definitions of individual rational, pareto optimal, and the negotiation set NS are standard definitions from game theory and bargaining theory (see, for example, [9, 11, 4]). This negotiation protocol is not general enough to be used in conflict situations because NS might be empty. Even in non-conflict situations NS can be empty. If all the states that satisfy (partially or fully) both goals are too expensive to reach, then the negotiation set may be empty even though it is not a conflict situation.
A more general negotiation protocol would be to negotiate over deals that are pairs of mixed joint plans.

Definition 8

- *A* Deal *is* (δ_A, δ_B, q) *where* δ_i *are mixed joint plans and* $0 \leq q \leq 1 \in \mathbb{R}$ *is the probability that the agents will perform* δ_A *(they will perform* δ_B *with probability* $1 - q$*).*
- $\text{Utility}_i(\delta_A, \delta_B, q) = q(\text{Worth}_i(\delta_i(s)) - Cost_i(\delta_i)) - (1-q)(\text{Worth}_i(\delta_j(s)) - Cost_i(\delta_j))$.

Theorem 1 *If for each* i*,* Worth_i *is a positive function, then* $NS \neq \emptyset$*.*

Proof. If Worth_i is a positive function then $U_i \geq 0$ because in the worst case $f_i = s$ (this means that $s \rightarrow f_i = \Lambda$); thus $U_i = \text{Worth}(s) - 0 \geq 0$. To show that $NS \neq \emptyset$ it is sufficient to show that there exists an individual rational deal. The deal $((s \rightarrow f_A, \Lambda)\colon 1, (\Lambda, s \rightarrow f_B)\colon 0, q)$ for any $0 \leq q \leq 1$ is individual rational because $\forall i$: $\text{Utility}_i((s \rightarrow f_A, \Lambda)\colon 1, (\Lambda, s \rightarrow f_B)\colon 0, q) = qU_i + (1-q)\text{Worth}_i(f_j) \geq 0$. □

When the Worth functions are not positive, then the negotiation set may be empty. If both agents' Worth functions have some (common) lower bound, we can normalize the two Worth functions and the Cost function by using a linear transformation, such that both Worth functions will be positive.

7 Examples of Worth Functions

Having presented the general framework for negotiation and goal relaxation, we shall now look at several domains and examine how each might dictate a different specific approach to the goal relaxation problem. These different approaches are manifested in alternate Worth functions for the various domains, because of domain-specific attributes as to what constitutes a goal.

7.1 Subgoals Set

An agent i may have a set of distinct subgoals or tasks $\{g_i^k \mid k = 1 \ldots n_i\}$ that he has to achieve. Each subgoal g_i^k's worth to him is W_i^k. In this case, his overall goal is a conjunction of subgoals, $g_i = \bigwedge_{k=1}^{n_i}(g_i^k)$. The worth of g_i for agent i is $\sum_{k=1}^{n_i}(W_i^k)$. The worth function in this case might be defined as

$$\text{Worth}(f) = \sum_{f \models g_i^k} (W_i^k).$$

Example: The Blocks World Domain
There is a table and a set of blocks. A block can be on the table or on some other block, and there is no limit to the height of a stack of blocks. However, on the table there are only a bounded number of slots into which blocks can be placed. There are two operations in this world: $\mathrm{PickUp}(i)$ — Pick up the top block in slot i (can be executed whenever slot i is not empty), and $\mathrm{PutDown}(i)$ — Put down the block which is currently being held into slot i. An agent can hold no more than one block at a time. Each operation costs 1. Let the initial state of the world be as shown in Figure 1.

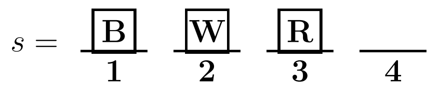

Figure 1: The Initial State of the World

Agent A has two subgoals: g_A^1 is "The Black box is clear at slot 2, but not on the table" and g_A^2 is "The White box is alone at slot 3". The worth of the two subgoals is $W_A^1 = 4$ and $W_A^2 = 6$. If Agent A were alone in the world, he would bring the world to one of the states in Figure 2.

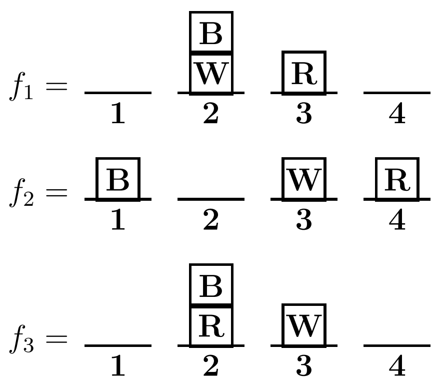

Figure 2: The Possible Final States

Agent A is indifferent between f_1, f_2 and f_3 because in all cases his utility is 2.

- $f_1 \models g_A^1$ therefore $\mathrm{Worth}_A(f_1) = 4$. $\mathrm{Cost}(s \rightarrow f_1) = 2$, therefore the utility of agent A from f_1 is $2 = (4 - 2)$.
- $f_2 \models g_A^2$ therefore $\mathrm{Worth}_A(f_2) = 6$. $\mathrm{Cost}(s \rightarrow f_2) = 4$, therefore the utility of agent A from f_2 is $2 = (6 - 4)$.
- $f_3 \models g_A^1 \cap g_A^2$ therefore $\mathrm{Worth}_A(f_3) = 10 = (6 + 4)$. $\mathrm{Cost}(s \rightarrow f_3) = 8$, therefore the utility of agent A from f_3 is $2 = (10 - 8)$.

Agent A is indifferent between f_1, f_2 and f_3 even though f_3 fully satisfies his goal, and f_1 and f_2 only partially satisfy his goal. This is due to the definition of Worth_A—an agent gets positive utility from achieving a subgoal, and zero utility from unachieved subgoals. There can be domains where an agent gets some penalty (negative utility) from unachieved subgoals. Let C_i^k be the penalty that agent i gets when his subgoal g_i^k is not achieved (it is convenient to think of penalties as *negative* numbers). We could then redefine the worth function to be

$$\text{Worth}(f) = \sum_{f \models g_i^k} W_i^k + \sum_{f \not\models g_i^k} C_i^k.$$

In the previous example, if $C_A^1, C_A^2 < 0$ then agent A prefers f_3 over f_1 and f_2.

Example: Let the initial state of the world be as in Figure 3.

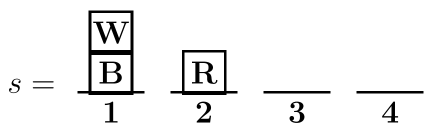

Figure 3: The Initial State of the World

- Agent A has two subgoals: g_A^1 is "The Black box is on the White box at slot 1" and g_A^2 is "The Red box is clear at slot 3". The worth of the two subgoals is $W_A^1 = 10$ and $W_A^2 = 4$. The penalties for unachieved goals are $C_A^1 = C_A^2 = 2$.
- Agent B has two subgoals: g_B^1 is "The Black box is on the White box at slot 1" and g_B^2 is "The Red box is clear at slot 4". The worth of the two subgoals are $W_B^1 = 10$ and $W_B^2 = 4$. The penalties for unachieved goals are $C_B^1 = C_B^2 = 2$.

The agents share one subgoal ($g_A^1 = g_B^1$) and have a conflict over the other subgoals. If each agent were alone in the world, in order to fully achieve his goal he would have a cost of at least $10 = (8 + 2)$, which would give him a utility of $U_i = 4 = (10 + 4) - 10$. Both goals cannot be fully achieved at the same time, but they can be partially achieved. The best joint plan T that achieves the swap in slot 1 (g_A^1 and g_B^1) costs 2 for each agent. They will agree on the deal $(\delta_A, \delta_B, 0.5)$ where the joint plan δ_i is the concatenation of T, and the one-agent plan where i moves the Red box in order to achieve g_i^2. The two possible final states can be seen in Figure 4.

$$\begin{aligned}
\forall i\ \text{Utility}_i(\delta_A, \delta_B, 0.5) &= 0.5(Worth_i(\delta_i(s)) - \text{Cost}_i(\delta_i) \\
&\quad +0.5(Worth_i(\delta_j(s)) - \text{Cost}_i(\delta_j) \\
&= 0.5((10+4) - (2+2)) + 0.5((10-2) - 2) \\
&= 8
\end{aligned}$$

$\forall i\ \text{Utility}_i(\delta_A, \delta_B, 0.5) > U_i$—even though the two goals can be only partially achieved, there exists a deal that is better for both agents than full achievement of their own goal by themselves.

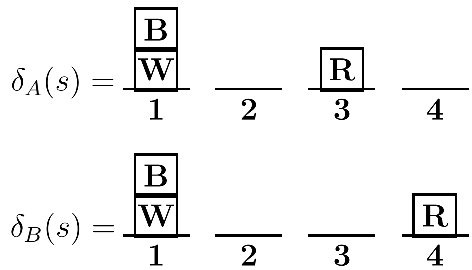

Figure 4: The Possible Final States

7.2 Distance Between States

One possible approach to the definition of the Worth function is to use some measurement of the distance of any state t from a state that achieves the full goal. We present here two examples showing different approaches to the measurement of this distance.

7.2.1 The One-Agent Plan Metric

Definition 9 *The distance between two world states* $s, f \in \mathrm{ST}$ *will be marked as* $d(s, f)$, *and will be defined as* $\mathrm{Cost}(s \to f)$.

The distance between state s and state f is the cost of the minimal one-agent plan that moves the world from state s to state f. This definition can be thought of (informally) as a *metric* over ST; it is reflexive and satisfies the triangle inequality.

- *Reflexivity* — $\forall t \in \mathrm{ST}\colon d(t, t) = 0$.
- *Triangle inequality* — $\forall s, t, f \in \mathrm{ST}\colon d(s, t) + d(t, f) \geq d(s, f)$.

This distance measure, however, is not necessarily symmetric:

- *Symmetric* — $\forall t, f \in \mathrm{ST}\ d(t, f) = d(f, t)$. This is true when talking about metrics, but in our domain it is not always true—nor do we need it to be true.

Using this distance measure, we can define the Worth function to be

$$\mathrm{Worth}(t) = W_i - \min_{f \models g_i}(d(t, f)).$$

7.2.2 The Delivery Problem

To illustrate the use of the above definition of Worth, we here introduce the Delivery Problem domain, which is an extension to the Postmen Problem introduced in [15].
There is a weighted graph $G = G(V, E)$. Each $v \in V$ represents a warehouse, and each $e \in E$ represents a road. The weight function $w\colon E \to \mathbb{R}$ is the distance of any given road. For each edge $e \in E$, $w(e)$ is the "length" of e or the "cost" of e. Each agent has to do some deliveries of containers between those warehouses. In order to do the deliveries,

agents can rent trucks. A truck can carry up to 4 containers. Each warehouse has a limited capacity.
The operations that can be done in this domain are:

- Load(c, t) — loads a container c on a truck t. The preconditions are:
  - Container c and truck t are at the same warehouse h.
  - Truck t has less then 4 containers on board.

  The results of the operation are:
  - Warehouse h has one container less.
  - Truck t has one container more.

  A Load operation costs 1.

- Unload(c, t) — unloads a container c from a truck t. The preconditions are:
  - Container c is on truck t.
  - Truck t is at some warehouse h.
  - Warehouse h is not full.

  The results of the operation are:
  - Warehouse h has one container more.
  - Truck t has one container less.

  The Unload operation costs 1.

- Drive(t, h) — Drives truck t to warehouse h. No preconditions for this operation. The result is that truck t is at warehouse h. The cost of this operation is equal to the distance (i.e., the minimal weighted path) between the current position of truck t and warehouse h.

There is another activity that needs to be taken into account: consumers come from time to time to take containers from the warehouses.
Agents can cooperate by using the same truck for delivery of both agents' containers. The only conflict that can occur is when the two agents need to deliver containers to the same warehouse, and there is not enough space for all the containers. In this case, one or both agents would not be able to fully achieve his goal; the goal may be completed later when some containers have been "consumed," i.e., removed from the warehouse by a consumer.
The one-agent plan metric is a reasonable heuristic measure to use when an agent whose goal has *not* been fully achieved can later achieve his whole goal. The one-agent distance as defined in this domain, for example, is an upper bound on how much the agent will need to spend to achieve his goal, when (or if) that achievement becomes possible.

Example: A Simple Delivery Problem

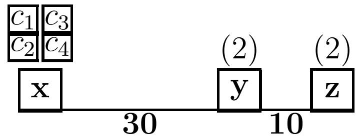

Figure 5: Simple Delivery Problem

Consider the graph in Figure 5. The weight (length) of each edge is written beside it. There is a place for only 2 more containers in warehouses y and z.
g_A is to deliver two containers c_1 and c_2 from x to z. g_B is to deliver container c_3 from x to y, and c_4 from x to z. $W_A = W_B = 50$.
If A were alone in the world, he could achieve his goal by renting a truck t_1 in warehouse x to take his 2 containers to z. The best one-agent plan to do this is $(\text{Load}(c_1, t_1), \text{Load}(c_2, t_1), \text{Drive}(t_1, z), \text{Unload}(c_1, t_1), \text{Unload}(c_2, t_1))$. This would give him a utility of $U_A = 50 - (1 + 1 + 40 + 1 + 1) = 6$.
If B were alone in the world he could achieve his goal by renting a truck t_2 in warehouse x to take his two containers to y and z. The best one-agent plan to do this is $(\text{Load}(c_3, t_2), \text{Load}(c_4, t_2), \text{Drive}(t_2, y), \text{Unload}(c_3, t_2), \text{Drive}(t_2, z), \text{Unload}(c_4, t))$.
This would give him a utility of $U_B = 50 - (1 + 1 + 30 + 1 + 10 + 1) = 6$.
The two goals cannot both be fully satisfied, because there isn't room for 3 additional containers in warehouse z. However, the agents can cooperate by using only one truck to make a partial delivery of the 4 containers.
The agents will agree to rent only one truck at x which will be loaded with the 4 containers—each will rent the truck with probability 0.5. The truck will unload container c_3 at y. c_4 or c_1 (each with probability 0.5) will also be unloaded at y. The truck will continue to z where the rest of the containers will be unloaded. Each agent has a 0.5 chance of fully achieving his goal. In both cases each agent is expected to pay half of the cost of the whole journey. If the agent's goal is only partially achieved, he will be at a "distance" of 12 from the achievement of his whole goal (one Load, a drive of length 10, and one Unload). The Utility for both agents from this deal is: $\text{Utility}_i = 0.5 \times (50 - 12) + 0.5 \times 50 - 0.5 \times 40 - 4 = 19$.

7.2.3 Probability Distance

Instead of using a metric over ST, we can define distance between world states and goals.

Definition 10 *The distance between a state $t \in \text{ST}$ and the goal g_i, signified as $d(t, g_i)$, is defined to be $1 - p$ where p is the probability that g_i will be achieved from state t.*

The Worth function in this case will be defined as

$$\text{Worth}(t) = (1 - d(t, g_i))W_i.$$

These definitions can be used in a domain that includes nondeterministic operations, that is, carrying out an action will send the world into one of several possible states, with some probability distribution. The existence of such operations would change our definition of a "plan" and the definition of a "joint plan" (for example, the schedule should be sensitive to the actual results of actions), which might result in a very different definition of a "deal." Discussion of this topic is beyond the scope of this article, and will be treated in future work.

8 Conclusion

We have presented a general solution to the negotiation problem in non-cooperative domains where agents can relax their initial goals (an agreement may lead to a situation in which one or both goals are only partially achieved). We presented a negotiation protocol that can be used in a general non-cooperative domain that includes goal relaxation. When goals can be relaxed, agents are negotiating both over what *parts* of their goals will be satisfied, and (in parallel) over the joint plan that will be implemented to satisfy those parts.

In a domain where goals can be relaxed, a specific Worth function over states should be defined, in order to provide a way of evaluating deals that only partially achieve an agent's goals. Several domain-dependent approaches can be used in defining the Worth function. We here presented several different definitions of the Worth function, including one for a domain in which a goal can be a set of independent subgoals, and one in which a conflict can be later resolved (and distance between world states is thus an appropriate measure).

Acknowledgments

This research has been partially supported by the Israel National Council on Research and Development, by the Leibniz Center for Computer Science at Hebrew University, and by the Center for Science Absorption, Office of Aliya Absorption, the State of Israel.

References

[1] Susan E. Conry, Robert A. Meyer, and Victor R. Lesser. Multistage negotiation in distributed planning. In Alan H. Bond and Les Gasser, editors, *Readings in Distributed Artificial Intelligence*, pages 367–384. Morgan Kaufmann Publishers, Inc., San Mateo, California, 1988.

[2] Edmund H. Durfee. *Coordination of Distributed Problem Solvers*. Kluwer Academic Publishers, Boston, 1988.

[3] Les Gasser. Social conceptions of knowledge and action: DAI foundations and open systems semantics. In *Proceedings of the Tenth International Workshop on Distributed Artificial Intelligence*, Bandera, Texas, October 1990.

[4] John C. Harsanyi. *Rational Behavior and Bargaining Equilibrium in Games and Social Situations*. Cambridge University Press, Cambridge, 1977.

[5] Sarit Kraus and Jonathan Wilkenfeld. The function of time in cooperative negotiations: Extended abstract. In *Proceedings of the Tenth Workshop on Distributed Artificial Intelligence*, Bandera, Texas, October 1990.

[6] Thomas Kreifelts and Frank v. Martial. A negotiation framework for autonomous agents. In *Proceedings of the Second European Workshop on Modeling Autonomous Agents and Multi-Agent Worlds*, pages 169–182, Saint-Quentin en Yvelines, France, August 1990.

[7] Kazuhiro Kuwabara and Victor R. Lesser. Extended protocol for multistage negotiation. In *Proceedings of the Ninth Workshop on Distributed Artificial Intelligence*, pages 129–161, Rosario, Washington, September 1989.

[8] Brigitte Laasri, Hassan Laasri, and Victor R. Lesser. Negotiation and its role in cooperative distributed problem problem solving. In *Proceedings of the Tenth International Workshop on Distributed Artificial Intelligence*, Bandera, Texas, October 1990.

[9] R. Duncan Luce and Howard Raiffa. *Games and Decisions*. John Wiley & Sons, Inc., New York, 1957.

[10] Thomas W. Malone, Richard E. Fikes, and M. T. Howard. Enterprise: A market-like task scheduler for distributed computing environments. In B. A. Huberman, editor, *The Ecology of Computation*. North-Holland Publishing Company, Amsterdam, 1988.

[11] John F. Nash. The bargaining problem. *Econometrica*, 28:155–162, 1950.

[12] Jeffrey S. Rosenschein and Michael R. Genesereth. Deals among rational agents. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 91–99, Los Angeles, California, August 1985.

[13] Reid G. Smith. *A Framework for Problem Solving in a Distributed Processing Environment*. PhD thesis, Stanford University, 1978.

[14] Katia P. Sycara. Resolving goal conflicts via negotiation. In *Proceedings of the Seventh National Conference on Artificial Intelligence*, pages 245–250, St. Paul, Minnesota, August 1988.

[15] Gilad Zlotkin and Jeffrey S. Rosenschein. Negotiation and task sharing among autonomous agents in cooperative domains. In *Proceedings of the The Eleventh International Joint Conference on Artificial Intelligence*, pages 912–917, Detroit, Michigan, August 1989. The International Joint Conference on Artificial Intelligence.

[16] Gilad Zlotkin and Jeffrey S. Rosenschein. Negotiation and task sharing in a non-cooperative domain. In *Proceedings of the Ninth Workshop on Distributed Artificial Intelligence*, pages 307–327, Rosario, Washington, September 1989.

[17] Gilad Zlotkin and Jeffrey S. Rosenschein. Blocks, lies, and postal freight: The nature of deception in negotiation. In *Proceedings of the Tenth International Workshop on Distributed Artificial Intelligence*, Bandera, Texas, October 1990.

[18] Gilad Zlotkin and Jeffrey S. Rosenschein. Negotiation and conflict resolution in non-cooperative domains. In *Proceedings of The National Conference on Artificial Intelligence*, pages 100–105, Boston, Massachusetts, August 1990. The American Association for Artificial Intelligence.