Rational Coordination in Multi-Agent Environments

Piotr J. Gmytrasiewicz* and Edmund H. Durfee†
*Department of Computer Science and Engineering
University of Texas at Arlington, Arlington, TX 76019-0015
piotr@cse.uta.edu
†Department of Electrical Engineering and Computer Science
University of Michigan
Ann Arbor, Michigan 48109
durfee@umich.edu

October 6, 1999

Abstract

We adopt the decision-theoretic principle of expected utility maximization as a paradigm for designing autonomous rational agents, and present a framework that uses this paradigm to determine the choice of coordinated action. We endow an agent with a specialized representation that captures the agent's knowledge about the environment and about the other agents, including its knowledge about their states of knowledge, which can include what they know about the other agents, and so on. This reciprocity leads to a recursive nesting of models. Our framework puts forth a representation for the recursive models and, under the assumption that the nesting of models is finite, uses dynamic programming to solve this representation for the agent's rational choice of action. Using a decision-theoretic approach, our work addresses concerns of agent decision-making about coordinated action in unpredictable situations, without imposing upon agents pre-designed prescriptions, or protocols, about standard rules of interaction. We implemented our method in a number of domains and we show results of coordination among our automated agents, among human-controlled agents, and among our agents coordinating with human-controlled agents.

[0]This research was supported, in part, by the Department of Energy under contract DG-FG-86NE37969, by the National Science Foundation under grant IRI-9015423, by the PYI award IRI-9158473, by ONR grant N00014-95-1-0775, and by the National Science Foundation CAREER award IRI-9702132.

1 Introduction

In systems involving multiple agents, system builders have traditionally analyzed the task domain of interest and, based on their analyses, imposed upon the agents certain rules (laws, protocols) that constrain the agents into interacting and communicating according to patterns that the designer deems desirable. Thus, research into coordination techniques has often led to prescriptions for task-sharing protocols, such as the Contract Net [73], for rules of interaction such as social laws [71], for negotiation conventions [67], and so on. The emphasis in this prior work has been to provide the agents with ready-to-use knowledge that guides their interactions, so that their coordination achieves certain properties desirable from the designer's point of view, such as conflict avoidance, stability, fairness, or load balancing.

The fundamental problem we address in this paper, on the other hand, is how agents should make decisions about interactions in cases where they have no common pre–established protocols or conventions to guide them.[1] Our argument is that an agent should rationally apply whatever it does know about the environment and about the capabilities, desires, and beliefs of other agents to choose (inter)actions that it expects will maximally achieve its own goals. While this kind of agent description adheres to the knowledge-level view (articulated by Newell [56]) that is a cornerstone of artificial intelligence, operationalizing it is a complex design process. Our work, as discussed in this paper, contributes to formalizing a rigorous, computational realization of an agent that can rationally (inter)act and coordinate in a multi-agent setting, based on knowledge it has about itself and others, without relying on protocols or conventions.

In our work, we use the normative decision-theoretic paradigm of rational decision-making under uncertainty, according to which an agent should make decisions so as to maximize its expected utility [17, 20, 23, 32, 40, 68]. Decision theory is applicable to agents interacting with other agents because of uncertainty: The abilities, sensing capabilities, beliefs, goals, preferences, and intentions of other agents clearly are not directly observable and usually are not known with certainty. In decision theory, expected utility maximization is a theorem that follows from the axioms of probability and utility theories [24, 53]. In other words, if an agent's beliefs about the uncertain environment conform to the axioms of probability theory, and its preferences obey the axioms of utility theory (see, for example, [68] page 474), then the agent should choose its actions so as to maximize its expected utility.[2].

The expected utilities of alternative courses of action are generally assessed based on their expected results. Intuitively, an agent is attempting to quantify how much better off it would be in a state resulting from it having performed a given action. In a multi-agent setting, however, an agent usually cannot anticipate future states of the world unless it can hypothesize the actions of other agents. Therefore, it may be beneficial for the agent to model

[1] We would like to stress that our approach does not forbid that agents interact based on protocols. However, to the extent that protocols specify the agent's action the agent does not need to deliberate about what to do and our approach is not applicable. If the protocol is not applicable or leaves a number of alternatives open then the agent needs to choose, and should do so in a rational manner.

[2] Some authors have expressed reservations as to the justifiability of these axioms. See the discussions in [50] and the excellent overview of descriptive aspects of decision theory in [10].

other agents influencing its environment to assess the outcomes and the utilities of its own actions. We say that an agent is *coordinating* with other agents precisely when it considers the anticipated actions of others as it chooses its own action.

An agent that is trying to determine what the other agents are likely to do may model them as rational as well, thereby using expected utility maximization as a descriptive paradigm.[3] This, in turn, leads to the possibility that they are similarly modeling other agents in choosing their actions. In fact, depending on the available information, this nested modeling could continue on to how an agent is modeling other agents that are modeling how others are modeling, and so on.

Thus, to rationally choose its action in a multi-agent situation, an agent should represent the, possibly nested, information it has about the other agent(s), and utilize it to solve its own decision-making problem. This line of thought, that combines decision-theoretic expected utility maximization with reasoning about other agent(s) that may reason about others, leads to a variant of game theory that has been called a decision-theoretic approach to game theory [3, 7, 41, 62]; we will compare it to traditional game theory further in Section 6. We would also like to remark that there are several ways in which the agent can avoid explicit representation and reasoning with all of its knowledge each time it needs to make a decision. Some of these methods of bounding the agent's rationality compile the available information into, say, condition-action rules, while some neglect information that cannot change, or is unlikely to change, the solution to the decision-making problem at hand. We outline some of these approaches in Section 7.

To help the reader put our work in perspective, we should stress that the representations we postulate here are only used for the purpose of decision-making in multi-agent situations; we do not postulate a general knowledge-representation and reasoning formalism. Thus, the representations we discuss are invoked only when there is a need for making a decision about which course of action to pursue, and our methods use many of the other components constituting a full-fledged autonomous agent. These usually include a suitably designed knowledge base containing a declarative representation of information about the environment and the other agents[4], sensing and learning routines that update the KB, planning routines that propose alternative courses of action, and so on. This paper does not address any of the difficult challenges posed by learning, sensing and planning; we concentrate solely on the issue of decision-making, understood as choosing among alternative courses of action enumerated, for example, by a symbolic planning system, using the information available in the knowledge base.

In the next section, we introduce an example of recursive modeling, while Section 3 formally presents the Recursive Modeling Method's (RMM) representation of nested knowledge and its solution concept. Section 4 illustrates the solution method through an example. We then report on a number of coordination experiments (Section 5), contrast RMM to other relevant

[3]The use of expected utility maximization to predict and explain human decision making is widely used in economics. See the overview in [10].

[4]Our implementation use a KB configured as an ontology of classes/frames and their instantiations extended to contain uncertain information [45].

work (Section 6), and discuss the complexity issues (Section 7). We conclude by summarizing RMM's contributions and open research problems (Section 8).

2 An Example of Recursive Modeling

The main goal of our method is to represent and reason with the relevant information that an agent has about the environment, itself, and other agents, in order to estimate expected utilities for its alternative courses of action, and thus to make a rational decision in its multi-agent situation. To choose an action that maximizes its individual utility, an agent should predict the actions of others. The fact that an agent might believe that other agents could be similarly considering the actions of others in choosing an action gives rise to the recursive nesting of models.

For the purpose of decision-making, RMM compactly folds together all of the relevant information an agent might have in its knowledge base, and summarizes the possible uncertainties as a set of probability distributions. This representation can reflect uncertainty as to the other agents' intentions, abilities, preferences, and sensing capabilities. Furthermore, on a deeper level of nesting, the agents may have information on how other agents are likely to view them, how they themselves think they might be viewed, and so on.

To facilitate the analysis of the decision-making behavior of the agents involved, the relevant information on each of the recursive levels of modeling is represented in RMM as a set of payoff matrices. In decision theory and game theory, payoff matrices have been found to be powerful and compact representations, fully summarizing the current contents of an agent's model of its external environment, the agent's capabilities for action in this environment, the relevant action alternatives of the other agents involved, and finally, the agent's preferences over the possible joint actions of the agents.

Given a particular multi-agent situation, a payoff matrix can be constructed from the information residing in the KB by various means. For example influence diagrams, widely used in the uncertainty in AI community, can be compiled into unique payoff matrices by summarizing the dependence of the utility of agent's actions on the environment and on others' actions. Other methods include equipping probabilistic or classical planners with multiattribute utility evaluation modules, as in the work reported in [32, 36], and in our early system called Rational Reasoning System (RRS) [29], which combined hierarchical planning with a utility evaluation to generate the payoff matrices in a nuclear power plant environment. A similar method of generating payoff matrices is used in the air-defense domain we report on in Section 5. Because, as we mentioned, RMM is independent of methods used to generate payoff matrices in a specific domain, we do not consider these issues in much depth in this paper.

To put our description of RMM in concrete terms, we consider a particular decision-making situation encountered by an autonomous outdoor robotic vehicle, called R_1 (Figure 1), attempting to coordinate its actions with another robotic vehicle, R_2. We assume that the vehicles' task is to gather as much information about their environment as possible, for example by moving to vantage points that command a wide view, while minimizing the cost of motion

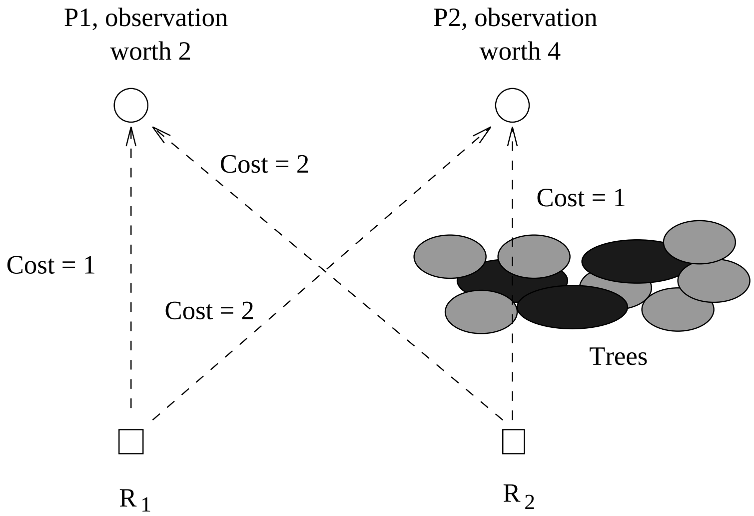

Figure 1: Example Scenario of Interacting Agents

itself. From the perspective of robot R_1, whose point of view we take in analyzing this situation, two possible vantage points P1 and P2 are worth considering. P2 has a higher elevation and would allow twice as much information to be gathered as P1, and so, the robot is willing to incur greater cost to go to P2. Based on domain-specific knowledge, in this example R_1 expects that gathering information at P2 is worth incurring a cost of 4 (or, put another way, the information gathered from P2 has an expected value of 4), while the observation from P1 is worth 2.

R_1 thus has three possible courses of action: it can move to P1 and gather information there (action a_1^1); it can move to P2 and gather information there (a_2^1); or it can do neither and just sit still (a_3^1).[5] The expected cost (time or energy) to R_1 of pursuing each of these courses of action is proportional to the distance traveled, yielding a cost of 1 for a_1^1, 2 for a_2^1, and 0 for a_3^1. We further assume in this example that each of the robots can make only one observation, and that each of them benefits from *all* information gathered (no matter by which robot), but incurs cost only based on its own actions.

Given that the above information resides in robot R_1's knowledge base, R_1 can build a payoff matrix that summarizes the information relevant to its decision-making situation. The relevant alternative behaviors of R_2 that matter are labeled a_1^2 through a_3^2, and correspond to R_2's alternative plans of taking the observation from point P1, P2, and staying put, respectively. Thus, the entry in the matrix corresponding to R_1's pursuing its option a_1^1 and R_2's pursuing a_2^2 is the payoff for R_1 computed as the total value of the information gathered by

[5]These courses of action could have been proposed as plausible by a symbolic planner, and each of them may have to be further elaborated by the robot. While all possible detailed plans for these high-level courses of action could be enumerated and represented in a payoff matrix, it is clearly desirable to include just a few abstract actions or plans.

both robots from both P1 and P2 minus R_1's own cost: (2+4) - 1 = 5. The payoff to R_1 corresponding to R_1's pursuing a_1^1 and R_2's pursuing a_1^2 is (2+0) - 1 = 1, since the information gathered is worth 2 and redundant observations add no value. All of the payoffs can be assembled in the payoff matrix depicted on top of the structure in Figure 2.

In order to arrive at the rational decision as to which of its three options to pursue, R_1 has to predict what R_2 will do. If R_2 were to take the observation from the point P2, i.e., its a_2^2 option, it would be best for R_1 to observe from P1. But if R_2 decided to stay put, R_1 should observe from the point P2, i.e., pursue its option a_2^1. In general, R_1 might be uncertain as to which action R_2 will take, in which case it should represent its conjecture as to R_2's action as a probability distribution over R_2's possible alternative courses of action. If R_1 thinks that R_2 attempts to maximize its own expected utility, then R_1 can adopt the intentional stance toward R_2 [19], treat R_2 as rational, and model R_2's decision-making situation using payoff matrices. R_2's payoff matrix, if it knows about both observation points, arrived at analogously to R_1's matrix above, has the form depicted in the middle branch in Figure 2.

That is not all, though, because R_1 realizes that robot R_2 possibly does not know about the observation point P2 due to the trees located between R_2 and P2. R_1, therefore, has to deal with another source of uncertainty: There is another alternative model of R_2's decision-making situation. If R_2 is unaware of P2, then it does not consider combinations of actions involving a_2^1 or a_2^2 and its payoff matrix is 2 $\times$ 2, as depicted in the left branch in Figure 2. The third model, in the right branch in Figure 2, represents the possibility that neither of the other two models of R_2's rational decision-making are correct. In this example, we assumed that R_1 does not have any other information that it could use to model R_2, and the third model is a no-information model. We elaborate on it further below.

R_1 can represent its uncertainty as to which of the models of R_2 is correct by assigning a subjective belief to each. In this example, we assume that R_1, having knowledge about the sensors available to R_2 and assessing the density of the foliage between R_2 and P2, assigns a probability 0.1 to R_2's being rational and seeing through the trees and a probability of 0.8 to it being rational but not being able to see P2. The remaining no-information model, which includes the possibility of R_2's being irrational, is assigned the probability of 0.1 (in [31, 74] we show how these models and their probabilities can be learned and updated based on the other agent's observed behavior).

Let us note that R_2's best choice of action, in each of the intentional models that R_1 has, also depends on what it, in turn, thinks that R_1 will do. Thus, R_1 should, in each of these models, represent what it knows about how R_2 models R_1. If it were to model R_1 as rational as well, the nesting of models would continue. R_2 might have some subjective probabilities over R_1's actions, based on a simplified model of R_1 or on past experiences with R_1. This would mean that the nesting terminates in what we call a *sub-intentional* model. If, on the other hand, R_2 were to lack the information needed to build a model of R_1's preferences over joint actions, then the nesting of models would terminate with other no-information models.

To keep this example simple and illustrative, let us make some arbitrary assumptions about how R_1's state of knowledge terminates, as follows: in the case that R_1 supposes that R_2 is of

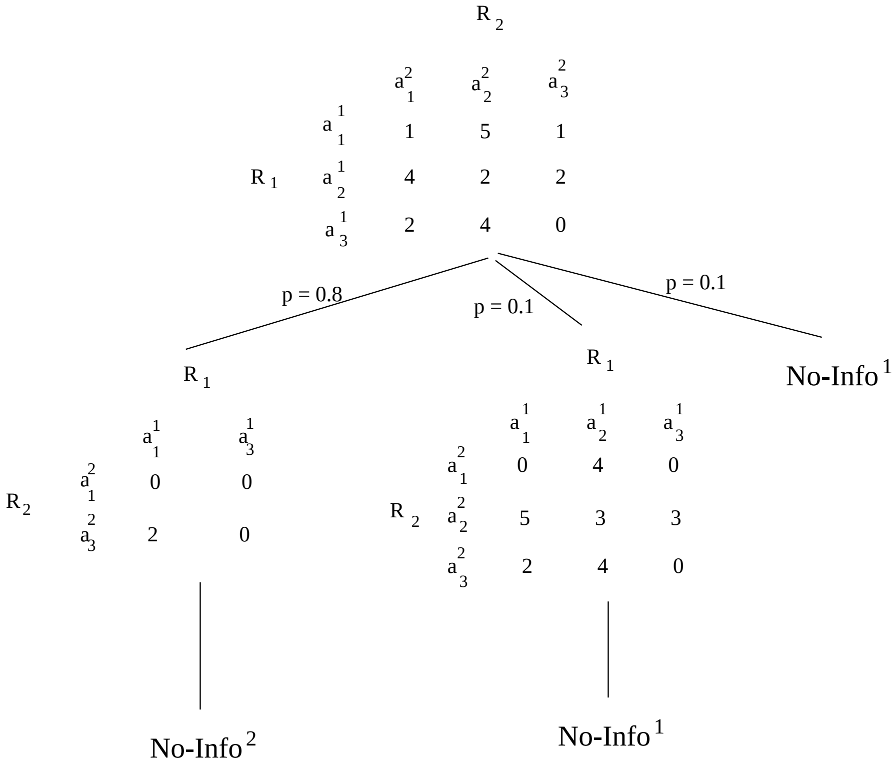

Figure 2: Recursive Model Structure depicting R_1's Decision-Making Situation in Example 1.

the type[6] that cannot see through the trees, then R_1 knows that R_2 does not know anything about R_1. But in the event that R_2 is of the type that can see through the trees, then R_1 itself has no knowledge in its knowledge base about how it might be modeled by R_2.

The scenario used here is relatively simple, but we invite the reader to develop his or her own intuitions at this point by considering the problem facing our robot R_1: What is the best course of action, given the information R_1 has about the situation and about R_2? Should R_1 move to P1 and hope that R_2 will cooperate by observing from P2? Or should R_1 go to P2 itself, due to the importance of this observation and in the face of uncertainty as to R_2's behavior? How does the probability of R_2's knowing about P2 influence R_1's choice? We provide the answers in Section 4.

The no-information models that terminate the recursive nesting in our example are at the leaves of the recursive model structure in Figure 2. These models represent the limits of the agents' knowledge: The model No-Info^2 represents the fact that, in the case when R_2 cannot see P2, R_1 knows that R_2 has no knowledge that would allow it to model R_1. Thus, the uncertainty is associated with R_2, and the model's superscript specifies that the state of no information is associated with its ancestor on the second level of the structure in Figure 2. The No-Info^1 model terminating the middle branch of the recursive structure represents R_1's own lack of knowledge (on the first level of the structure) of how it is being modeled by R_2, if

[6]Our use of this term coincides with the notion of agent's type introduced by Harsanyi in [37].

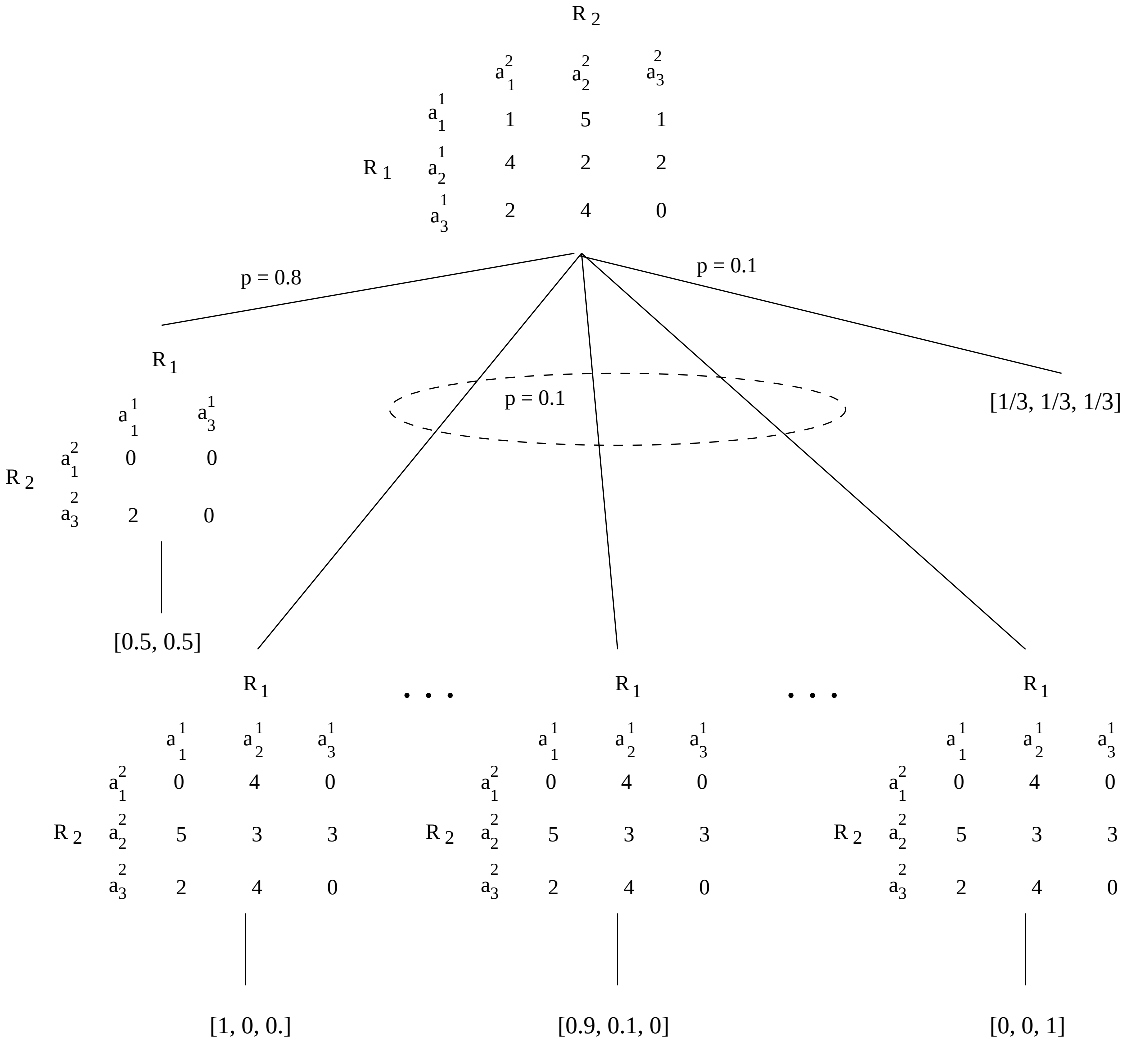

Figure 3: Semantics of the No-Information Models in Example 1.

R_2 can see through the trees. In general, the no-information models can represent knowledge limitations on any level; the limitations of R_1's own knowledge,[7] R_1's knowing the knowledge limitations of other agents, and so on.

Figure 3 illustrates the semantics of the no-information models depicted in Figure 2. The no-information model No-Info^1 in the middle branch means that R_1 has no information about how it is modeled by R_2. Therefore, all of the conjectures that R_2 may have about R_1's behavior are possible and, according to the principle of indifference [16, 55], equally likely. This can be represented as the branch on the first level of the recursive structure splitting into infinite sub-branches, each of which terminates with a different, legal probability distribution describing R_2's conjecture about R_1's behavior. Cumulative probability of all of the sub-branches remains the same (0.1 in this example).[8] According to this interpretation, the

[7]Note that we assume the agent can introspect. This amounts to the agent's being able to detect the lack of statements in its knowledge base that describe beliefs nested deeper than the given level.

[8]Our representation here is related to the problem of "only knowing", discussed in [33, 48] and related

No-Info[1] model in the middle branch in Figure 2 is simply a shorthand notation for the more explicit representation depicted in Figure 3.

The no-information model No-Info[1] in the right branch in Figure 2 expresses the fact that R_1 has no other information based on which it could predict R_2's behavior. Again, this translates into all of the legal 2-vector distributions emanating from the model on the first level being possible and equally likely. It can be shown (see the principle of interval constraints discussed in [55]) that the set of all of these legal distributions can be equivalently represented by a uniform distribution over R_2's possible actions, a_1^2, a_1^2 and a_3^2, themselves: $[1/3, 1/3, 1/3]$. This distribution precisely represents R_1's lack of knowledge in this case, since its information content is zero.

The no-information model No-Info[2] in the left branch in Figure 2 is similar but more complicated. It expresses the fact that R_1 knows that if R_2 cannot see P2 then R_2 has no information based on which it could predict R_1's behavior. This translates into all of the legal 2-vector distributions, now emanating from the model on the second level, being possible and equally likely. It can be equivalently represented by a uniform distribution over R_1's possible actions, a_1^1 and a_3^1: $[0.5, 0.5]$, as depicted in Figure 3. The above interpretations keep an intuitive convention that branching due to uncertainty emanates from the model of the agent that is uncertain. The middle and right branches terminate with no-information models due to lack of knowledge of agent R_1, while the left branch terminates with No-Info[2] because we assumed that R_1 knows that R_2 has no information.

3 General Form of the Recursive Modeling Method

In this section, we formalize the intuitions behind the recursive modeling that we developed in the preceding section. The Recursive Modeling Method consists of the recursive model structure that contains information the agent has in all of its nested levels, and the solution method that uses dynamic programming to arrive at the rational choice of an agent's action in a multi-agent situation. The reader who wishes to skip the formalism for now can proceed to Section 4 where we solve the example interaction.

3.1 Representation

We first formally define the payoff matrix, which is the basic building block of RMM's modeling structure. A payoff matrix represents the decision-making situation an agent finds itself in when it must choose an action to take in its multi-agent environment. Following the definition used in game theory [63], we define the payoff matrix, P_{R_i}, of an agent R_i as a triple $P_{R_i} = (R, A, U)$.

R is a set of agents in the environment, labeled R_1 through R_n ($n \geq 1$). R includes all decision-making agents impacting the welfare of the agent R_i.

references.

A is defined as a cross product: $A = A_1 \times A_2 \times \cdots \times A_n$, where set $A_j = \{a_1^j, a_2^j \cdots\}$ represents the alternative actions of agent R_j. The elements of A are the *joint moves* of the n agents in question. We additionally define a *joint move of the other agents* as an element of the following set: $A_{-i} = A_1 \times A_2 \times \cdots \times A_{i-1} \times A_{i+1} \times \cdots \times A_n$. The joint move of the other agents specifies the moves of all of the agents except R_i. We further demand that the sets of alternative actions of the agents be exhaustive, and that the alternatives be mutually exclusive.

Finally, U is a payoff function that assigns a number (expected payoff to R_i) to each of the joint actions of all of the agents: $U : A \longrightarrow \mathbf{R}$, where $\mathbf{R}$ is the set of real numbers. Intuitively, a purposeful agent has reason to prefer some actions (that further its purposes in the current situation) to others [78]. Our ability to represent agents' preferences over actions as payoffs follows from the axioms of utility theory, which postulate that ordinal preferences among actions in the current situation can be represented as cardinal, numeric values (see [13, 20] for details). We represent R_i's payoff associated with a joint action $(a_k^1, \cdots, a_m^i, \cdots, a_l^n)$ as $u^{R_i}_{a_k^1 \cdots a_m^i \cdots a_l^n}$.

We now define the recursive model structure of agent R_i, RMS_{R_i}, as the following pair:

$$RMS_{R_i} = (P_{R_i}, RM_{R_i}), \tag{1}$$

where P_{R_i} is R_i's payoff matrix, and RM_{R_i} is R_i's recursive model, which summarizes the information R_i has about the other $n-1$ agents in the environment. A recursive model RM_{R_i} is defined as a probability distribution over the alternative models of the other agents. Thus, if $M^{(R_i,\alpha)}$ is taken to denote the α-th of R_i's alternative models of the other agents, i.e., all agents except R_i, then R_i's recursive model assigns to it a probability, $p_\alpha^{R_i}$. These probabilities represent R_i's subjective belief that each of the alternative models is correct. We call $p_\alpha^{R_i}$'s *modeling* probabilities. They sum to unity: $\sum_{\alpha=1}^m p_\alpha^{R_i} = 1$ (for the example in Figure 2 they are the probabilities of the three branches.) To make our exposition more transparent we have assumed above that the set of alternative models is finite, but one could generalize the modeling probability to be defined over a measurable infinite space of alternative models.[9] Each of the alternative models of the other agents is a list of models of each of the agents:

$$M^{(R_i,\alpha)} = (M_{R_1}^{(R_i,\alpha)}, ..., M_{R_{i-1}}^{(R_i,\alpha)}, M_{R_{i+1}}^{(R_i,\alpha)}, .., M_{R_n}^{(R_i,\alpha)}). \tag{2}$$

The models $M_{R_j}^{(R_i,\alpha)}$, that R_i can have of R_j, come in three possible forms:

$$M_{R_j}^{(R_i,\alpha)} = \begin{cases} Intent_{R_j}^{(R_i,\alpha)} & \text{– the intentional model,} \\ No-Info_{R_j}^{(R_i,\alpha),\phi} & \text{– the level-}\phi\text{ no-information model,} \\ Sub-Int_{R_j}^{(R_i,\alpha)} & \text{– the sub-intentional model.} \end{cases} \tag{3}$$

The intentional model corresponds to R_i modeling R_j as a rational agent. It is defined as:

[9] In the next subsection we show how an infinite space of models can be transformed into an equivalent finite set.

$$Intent_{R_j}^{(R_i,\alpha)} = RMS_{R_j}^{(R_i,\alpha)}, \quad (4)$$

that is, it is the recursive model structure that agent R_i ascribes to agent R_j. This structure, as defined in Equation 1, further consists of the payoff matrix that R_i ascribes to R_j in this model, $P_{R_j}^{(R_i,\alpha)}$, and the recursive model $RM_{R_j}^{(R_i,\alpha)}$ containing the information R_i thinks R_j has about the other agents. For example, the two models in the left and the middle branches in Figure 2 are intentional models.

The level-ϕ no-information model, No-Info$_{R_j}^{(R_i,\alpha),\phi}$, represents the limits of knowledge associated with the agent modeled on the ϕ level of nesting in the branch the no-information model resides in. In other words, No-Info$_{R_j}^{(R_i,\alpha),\phi}$ located on a level l, represents R_i's belief that the agent modeled on level ϕ has run out of knowledge at level l of R_i's modeling structure. According to this semantics, the superscript of the no-information model has to be between 1 (corresponding to the agent R_i running out of information, as in Figure 2) and a value one less than the level on which the no-information model is located in the recursive structure. Thus, for a no-information model, No-Info$^{\phi}$, located on level l, we have: $1 \leq \phi \leq l-1$.

The no-information models assign uniform probabilities to all of the alternative distributions over the actions of the other agents and contain no information [55] beyond the currently considered level of nesting, representing the limits of knowledge reached at a particular stage of recursive modeling. The use of no-information models in our decision-making framework reflects a situation in which a symbolic KB of the agent in question contains the agent's beliefs about the others' beliefs nested to some level, but it does not contain any information nested deeper, for example because the agent did not have a chance to acquire any more information.

The sub-intentional model is a model which does not include the ascription of beliefs and preferences, and does not use rationality to derive behavior.[10] Besides the intentional stance, Dennett [19] enumerates two sub-intentional stances: The *design* stance, which predicts behavior using functionality (such as how the functions of a console controller board's components lead to its overall behavior [35]), and the *physical stance*, which predicts behavior using the description of the state of what is being modeled along with knowledge of its dynamics (like in the qualitative model of a bouncing ball [25], or finite state automata models in [11]). These models can be useful for an agent that can incorporate techniques such as model-based reasoning or qualitative physics to make predictions about the behavior of sub-intentional[11] entities, resulting in a probability distribution over their alternative behaviors, as enumerated in the agent's payoff matrix. Further, any informative conjecture, i.e., a probability distribution over others' actions, can be treated as a sub-intentional model, if it has been arrived at without ascribing rationality to the modeled entity. For example, a conjecture as to another's actions may be derived from plan recognition, from past actions (as in [39]), or from information related by a third agent, and it can be given a probabilistic weight according to the assessment of its faithfulness within the RMM framework.

[10]According to Dennett [19], such a sub-intentional agent does not even satisfy the basic requirement of agenthood. It is simply an entity, then, rather than an agent proper.

[11]That is, not following the decision-theoretic principles of rationality.

The definition of the recursive model structure and the intentional model are recursive, but, as we argue in more detail later, it is likely to be finite due to practical difficulties in attaining infinitely nested knowledge. In other words, in representing the content of its KB about its own decision-making situation, the situations of the other agents, and of what the other agents know about others, the agent is likely to run out of knowledge at some level of nesting, in which case the recursion terminates with a level-1 no-information model. Of course some recursive branches can also terminate with higher level no-information models representing the possible limitations of other agents' knowledge, or with sub-intentional models that do not lead to further recursion. Thus, the no-information models are not intended as an *ad hoc* means to terminate the recursive structure of models. Rather, in our knowledge-based view, the branches of the recursive structure terminate with a no-information model when, and only when, the limits of the agents' knowledge, contained in its KB, are reached. In that way, all of the agent's knowledge relevant to the decision-making process is used to derive the rational coordinated choice of action.

3.2 Solving RMM Using Dynamic Programming

The recursive nature of RMM makes it possible to express the solution to the problem of choice that maximizes expected utility on a given level of modeling in terms of the solutions to choices of the agents modeled on deeper levels. Thus, to solve the optimization problem on one level requires solutions to subproblems on the lower level. This means that the problem exhibits *optimal substructure* [18], and that a solution using dynamic programming can be formulated. The solution traverses the recursive model structure propagating the information bottom-up. The result is an assignment of expected utilities to the agent's alternative actions, based on all of the information the agent has at hand about the decision-making situation. The rational agent can then choose an action with the highest expected utility.

Clearly, the bottom-up dynamic programming solution requires that the recursive model structure be finite and terminate. Thus, we make the following assumption:

Assumption 1: *The recursive model structure, defined in Equation 1, is finite and terminates with sub-intentional or no-information models.*

We should remark that the assumption above complements an assumption that the agents possess infinitely nested knowledge, called common knowledge or mutual knowledge, frequently made in AI and in traditional game theory. These two assumptions lead to two solution concepts; one discussed here, which is decision-theoretic and implemented with dynamic programming, the other one based on the notion of equilibria. We discuss the justifiability of these assumption further in Section 5.

The expected utility of the m-th element, a^i_m, of R_i's set of alternative actions is evaluated as:

$$u^{R_i}_{a^i_m} = \sum_{(a^1_k,\ldots,a^n_p)\in A_{-i}} p^{R_i}_{a^1_k\ldots a^n_p} u^{R_i}_{a^1_k\cdots a^i_m\cdots a^n_p} \tag{5}$$

where $p^{R_i}_{a^1_k\ldots a^n_p}$ represents R_i's conjecture as to the joint actions of the other agents, i.e., it is an element of the probability distribution over the set of joint moves of the other agents A_{-i}.

We refer to $p^{R_i}_{a^1_k \cdots a^n_p}$ as *intentional* probabilities. $u^{R_i}_{a^1_k \cdots a^i_m \cdots a^n_p}$ is R_i's expected payoff residing in its payoff matrix, P_{R_i}.

R_i can determine the intentional probabilities $p^{R_i}_{a^1_k \cdots a^n_p}$ by using its modeling knowledge of other agents contained in the recursive model RM_{R_i}. As defined in the preceding section, R_i can have a number of alternative models $M^{(R_i,\alpha)}$ of the other agents, and a modeling probability, $p^{R_i}_\alpha$, associated with each of them. If we label the predicted probability of joint behavior of the other agents resulting from a model $M^{(R_i,\alpha)}$ as $p^{(R_i,\alpha)}_{a^1_k \cdots a^n_p}$, we can express the overall intentional probability of the other agents' joint moves, $p^{R_i}_{a^1_k \cdots a^n_p}$, as an average over all possible models (this is known as Bayesian model averaging [38]):

$$p^{R_i}_{a^1_k \cdots a^n_p} = \sum_\alpha p^{R_i}_\alpha \times p^{(R_i,\alpha)}_{a^1_k \cdots a^n_p}. \tag{6}$$

The joint probability, $p^{(R_i,\alpha)}_{a^1_k \cdots a^n_p}$, of the other agents' behaviors resulting from a model $M^{(R_i,\alpha)}$, can in turn be expressed as a product of the intentional probabilities for each of the agents individually, $p^{(R_i,\alpha)}_{a^j_k}$, resulting from a model $M^{(R_i,\alpha)}_{R_j}$:

$$p^{(R_i,\alpha)}_{a^1_k \cdots a^n_p} = p^{(R_i,\alpha)}_{a^1_k} \times \cdots \times p^{(R_i,\alpha)}_{a^n_p} \tag{7}$$

If the model $M^{(R_i,\alpha)}_{R_j}$ is in the form of a sub-intentional model, then the probabilities $p^{(R_i,\alpha)}_{a^j_k}$ indicating the expected behavior of the entity can be derived by whatever techniques (statistical, model-based, qualitative physics, etc.) R_i has for predicting behavior of such entities.

If R_i has assumed an intentional stance toward R_j in its model $M^{(R_i,\alpha)}_{R_j}$, i.e., if it is modeling R_j as a rational agent, then it has to model the decision-making situation that agent R_j faces, as specified in Equations 3 and 4, by R_j's payoff matrix $P^{(R_i,\alpha)}_{R_j}$ and its recursive model $RM^{(R_i,\alpha)}_{R_j}$. R_i can then identify the intentional probability $p^{(R_i,\alpha)}_{a^j_k}$ as the probability that the k-th alternative action is of the greatest utility to R_j in this model:

$$p^{(R_i,\alpha)}_{a^j_k} = Prob(u^{(R_i,\alpha),R_j}_{a^j_k} = Max_{k'}(u^{(R_i,\alpha),R_j}_{a^j_{k'}})). \tag{8}$$

$u^{(R_i,\alpha),R_j}_{a^j_{k'}}$ is the utility R_i estimates that R_j assigns to its alternative action $a^j_{k'}$ in this model, and it can be further computed as:

$$u^{(R_i,\alpha),R_j}_{a^j_{k'}} = \sum_{(a^1_o,\ldots,a^n_r)\in A_{-j}} p^{(R_i,\alpha),R_j}_{a^1_o \cdots a^n_r} u^{(R_i,\alpha),R_j}_{a^1_o \cdots a^j_{k'} \cdots a^n_r} \tag{9}$$

This equation is analogous to Equation 5 except it is based on R_i's model of R_j. The $u^{(R_i,\alpha),R_j}_{a^1_o \cdots a^j_{k'} \cdots a^n_r}$ are R_j's payoffs in the payoff matrix $P^{(R_i,\alpha)}_{R_j}$. The intentional probabilities $p^{(R_i,\alpha),R_j}_{a^1_o \cdots a^n_r}$ are what R_i thinks R_j assigns to other agents' actions. The probabilities $p^{(R_i,\alpha),R_j}_{a^1_o \cdots a^n_r}$ can in turn be expressed in terms of the models that R_i thinks R_j has of the other agents in the environment, contained in $RM_{R_j}^{(R_i,\alpha)}$, and so on.

The intentional stance R_i uses to model R_j is formalized in Equation 8. It states that agent R_j is an expected utility maximizer and, therefore, its intention can be identified as a course of action that has the highest expected utility, given R_j's beliefs about the world and its preferences.

What the intentional stance does not specify is how R_j will make its choice if it finds that there are several alternatives that provide it with the maximum payoff. Using the principle of indifference once more, R_i assigns an equal, nonzero probability to R_j's option(s) with the highest expected payoff, and zero to all of the rest.[12] Formally, we can construct the set of R_j's options that maximize its utility:

$$Amax_j^{(R_i,\alpha)} = \{a_k^j \mid a_k^j \in A_j^{(R_i,\alpha)} \wedge u_{a_k^j}^{(R_i,\alpha),R_j} = Max_{k'}(u_{a_{k'}^j}^{(R_i,\alpha),R_j})\}. \tag{10}$$

Then, the probabilities are assigned according to the following:

$$p_{a_k^j}^{(R_i,\alpha)} = \begin{cases} \frac{1}{|Amax_j^{(R_i,\alpha)}|} & \text{if } a_k^j \in Amax_j^{(R_i,\alpha)} \\ 0 & \text{otherwise.} \end{cases} \tag{11}$$

Finally, if R_i's model terminates with a no-information model, two cases arise. The first occurs when we have a no-information model No-Info$^\phi$ located on level $\phi + 1$ describing the limits of knowledge possessed by the agent modeled on level ϕ. This model is a shorthand for all legal distributions being possible and equally likely.[13] As could be expected, it can be shown (see the principle of interval constraints method in [55]) that it can be equivalently represented by a uniform distribution over the other agents' possible actions at this level, yielding the probabilities $p_{a_k^j}^{(R_i,\alpha)} = \frac{1}{|A_j|}$ specified in this model. The models in the leaves of the right- and left-most branches in Figure 3 illustrate this case.

The second, more complex case, occurs when a model No-Info$^\phi$ is located on level deeper than $\phi+1$. In this case we note that Equation 8 and Equation 9 define a finite number of equivalence classes among the infinite sub-branches represented by these no-information models. Namely, an intentional probability distribution used in Equation 9 to compute the intentional probabilities higher up the recursive model in Equation 8 is equivalent to another such distribution, provided that it also favors the same alternatives chosen as optimal in Equation 8. It follows that the no-information model in this case can be equivalently represented by a finite number ($|A_j|$ at most) of discrete branches, each representing such an equivalence class. The resulting discrete branches have a modeling probability, associated with the equivalence classes they represent, defined on the measurable space of possible intentional probability distributions in the leaves of the sub-branches. These branches can be terminated with any of

[12]As we mentioned, we use the expected utility maximization as a descriptive tool. See also [9, 10].

[13]The principle of indifference is applied here to the probability itself. See, for example, the discussion in [16] Section 1.G.

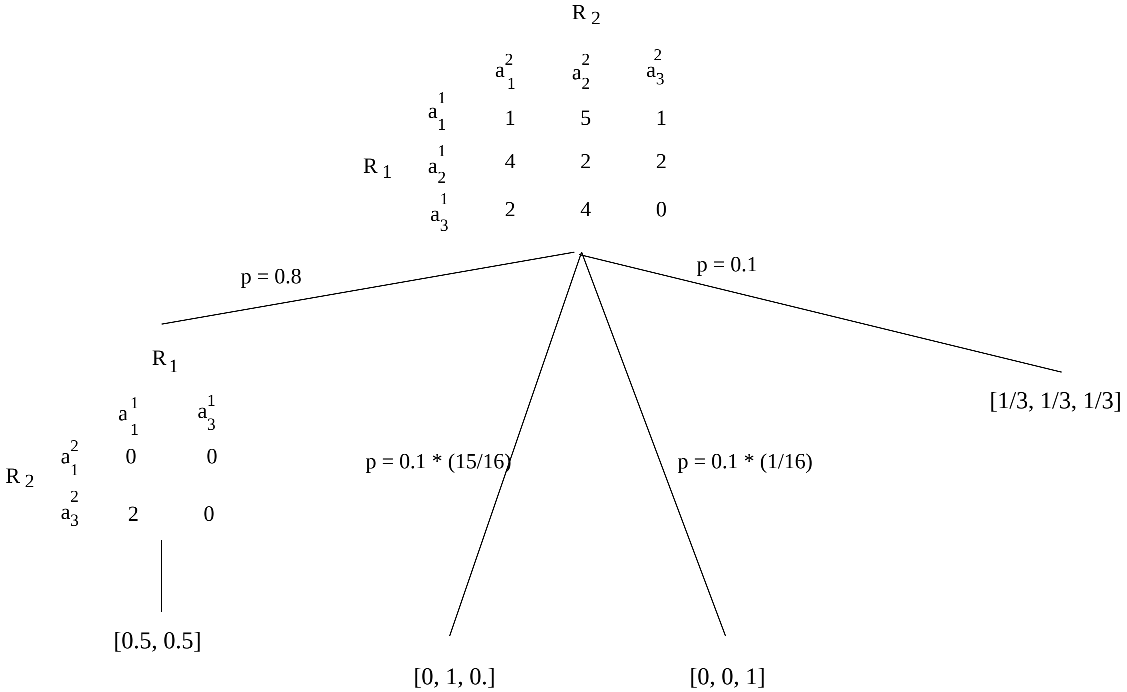

Figure 4: Transformed Recursive Model Structure for Example 1

these equivalent distributions on the level $\phi + 1$, or simply with the resulting probability distribution computed in Equation 8 on level ϕ. The information contained in these branches can then propagated upwards directly. We provide examples of these calculations in the following section.

4 Solving the Example Interaction

In this section, we solve the example decision-making problems presented in Section 2. We begin by replacing the infinite branching of the middle model of R_2 in Figure 3 with a finite number of equivalence classes. Note that some of the probability triples in the sub-branches in Figure 3, when used to calculate the expected utilities of R_2's actions in the matrix above make R_2's action a_2^2 the most preferable, while other triples may favor other actions. For example, if R_2 models R_1's expected behavior using the probability distribution $[1, 0, 0]$ over the actions a_1^1, a_2^1, and a_3^1, then, given R_2's payoff matrix, the expected utilities of R_2's alternatives a_1^2, a_2^2, and a_3^2, according to Equation 5 are 0, 5, and 2, respectively, and the action a_2^2 is preferred for R_2. Another distribution, say, $[0.9, 0.1, 0]$ also favors a_2^2 and thus belongs to the same equivalence class as $[1, 0, 0]$. The distribution $[0.1, 0.9, 0]$, on the other hand, makes the action a_3^2 preferable for R_2, and belongs to a different equivalence class.

The modeling probability of the branch representing the class favoring the action a_2^2 is computed as the proportion of all of the 3-vectors in Figure 3 that favor a_2^2, among all of the legal distributions over the three actions of R_1, $[x_1, x_2, x_3]$, such that $x_1 + x_2 + x_3 = 1$ and $0 \leq x_i \leq 1$, for i equal to 1 through 3. All of these legal 3-vectors form a triangular part of a plane in the three dimensional space spanned by the axes x_1, x_2, and x_3. The area of this triangle can be computed [51] as $\sqrt{3}\int_0^1\int_0^{1-x_1} dx_2 dx_1 = \frac{\sqrt{3}}{2}$. The part of the area of the legal 3-vectors that favor the action a_2^2, can be computed (again, see [51]) as $\sqrt{3}[\int_0^{0.25}\int_0^{0.75} dx_2 dx_1 + \int_{0.25}^{1}\int_0^{1-x_1} dx_2 dx_1] = \sqrt{3} \times \frac{15}{32}$, and the area that favors a_3^2 can be computed as $\sqrt{3}\int_{0.75}^{1}\int_0^{1-x_2} dx_1 dx_2 = \frac{\sqrt{3}}{32}$.[14] Thus, two equivalence classes among the 3-vectors that favor a_2^2 and a_3^2 have probabilities equal to $\frac{15}{16}$ and $\frac{1}{16}$, respectively, and these are the only classes that have a nonzero probability. Now, all of the sub-branches that favor each of the separate alternative actions can be lumped into a sub-branch ending with a single representative probability 3-vector favoring this particular action on level 3, or simply with the intentional distribution reflecting the favored action of agent R_2 on level 2. The resulting recursive structure for our example is depicted in Figure 4.

The recursive structure in Figure 4 can be solved with dynamic programming, which after reaching the bottom of the structure, propagates the results upwards as follows. The intentional probability distribution in the leftmost leaf in Figure 4—representing R_1's knowing that R_2 has no information about how to model R_1's intentions—is: $p_{R_1}^{(R_1,1),R_2} = [p_{a_1^1}^{(R_1,1),R_2}, p_{a_3^1}^{(R_1,1),R_2}] = [0.5, 0.5]$. Given R_2's payoff matrix in this case, the expected utilities of its alternatives in this model are computed (Equation 9) as the probabilistic sum of the payoffs:

$$
\begin{aligned}
u_{a_1^2}^{(R_1,1),R_2} &= p_{a_1^1}^{(R_1,1),R_2} \times 0 + p_{a_3^1}^{(R_1,1),R_2} \times 0 = 0 \\
u_{a_3^2}^{(R_1,1),R_2} &= p_{a_1^1}^{(R_1,1),R_2} \times 2 + p_{a_3^1}^{(R_1,1),R_2} \times 0 = 1
\end{aligned}
$$

Since the set of R_2's alternatives that maximize its expected payoff in this model has only one element ($Amax_2^{(R_1,1)} = \{a_3^2\}$) the probability distribution over the actions of agent R_2 in this model (Equations 10 and 11) is: $p_{R_2}^{(R_1,1)} = [p_{a_1^2}^{(R_1,1)}, p_{a_3^2}^{(R_1,1)}] = [0, 1]$. Thus, R_1 knows that if R_2 cannot see point P2 it will remain stationary.

The probability distributions over R_2's alternatives in the remaining three branches specify that R_2 will move toward P2 and make an observation from there (this is the case when R_2 can see P2 and its model of R_1 indicates that pursuing P2 is better), with the probability of $0.1 \times (15/16) = 0.09375$. With the probability $0.1 \times (1/16) = 0.00625$, R_2 will remain still even though it knows about P2, since its model of R_1 indicates that R_1 is likely to pursue observation from P2. The remaining no-information model has a probability of 0.1 and assigns equal probabilities to all of R_2's alternative actions.

[14] We found the method of logic sampling to be an effective approximate way to compute the values of the integrals here.

The four alternative models of R_2's behavior can be combined into the overall intentional distribution over R_2's actions as a probabilistic mixture of R_2's intentions in each of the alternative models (Equation 6): $p_{R_2}^{R_1} = [p_{a_1^2}^{R_1}, p_{a_2^2}^{R_1}, p_{a_2^2}^{R_1}] = .8 \times [0,0,1] + 0.09375 \times [0,1,0] + 0.00625 \times [0,0,1] + 0.1 \times [1/3,1/3,1/3] = [0.0333, 0.1271, 0.8396]$.

The expected utilities of R_1's alternative actions in its own decision-making situation (top matrix in Figure 4) can now be computed (Equation 5) as:

$$
\begin{aligned}
u_{a_1^1}^{R_1} &= 0.0333 \times 1 + 0.1271 \times 5 + 0.8396 \times 1 = 1.5084 \\
u_{a_2^1}^{R_1} &= 0.0333 \times 4 + 0.1271 \times 2 + 0.8396 \times 2 = 2.0666 \\
u_{a_3^1}^{R_1} &= 0.0333 \times 2 + 0.1271 \times 4 + 0.8396 \times 0 = 0.575
\end{aligned}
$$

Thus, the best choice for R_1 is to pursue its option a_2^1, that is, to move toward point P2 and make an observation from there. It is the rational coordinated action given R_1's state of knowledge, since the computation included all of the information R_1 has about agent R_2's expected behavior. Intuitively, this means that R_1 believes that R_2 is so unlikely to go to P2 that R_1 believes it should go there itself.

Let us note that the traditional tools of equilibrium analysis do not apply to this example since there is no common knowledge. However, the solution obtained above happens to coincide with one of two possible solutions that could be arrived at by traditional game-theoretic equilibrium analysis, if a number additional assumptions about what the agents know were made in this case. Thus, if R_1 were to assume that R_2 knows about the point P2, and that R_2 knows that R_1 knows, and so on, then R_1's move toward P2 would a part of the equilibrium in which R_1 goes to P2 and R_2 goes to P1. This shows that the solutions obtained in RMM analysis can coincide with game-theoretic solutions, but that it depends on fortuitous assumptions about agents' knowledge. It is also not difficult to construct a finite state of R_1's knowledge that would result in R_1's rational action to be pursuing observation from P1 and expecting R_2 to observe from P2, which happens to be the other equilibrium point that could be derived if the agents were assumed to have common knowledge [3] about P2. The coincidence would, again, be a matter of making *ad hoc* assumptions about the agents' states of knowledge.

5 Application Domains and Experiments

RMM fills a unique niche among multi-agent reasoning techniques based on pre-established protocols in many realistic domains for two main reasons. First, in many domains the environment is too variable and unpredictable for pre-established protocols to remain optimal in circumstances that could not be foreseen by the designers. Second, frequently, the group of interacting agents is not specified before hand[15], and one cannot rely on the agents' knowing

[15]This means that the multi-agent system is *open*.

which, if any, protocol to follow. Examples include numerous human-machine coordination tasks, such as many realistic space applications, in which robots need the ability to interact with both other robots and humans, as well as applications in defense-related domains, characterized by their inherently unpredictable dynamics. Other examples include telecommunications networks, flexible manufacturing systems, and financial markets.

In our work, we have looked more closely at applying RMM to coordinate autonomous manufacturing units [30], and applications to coordination and intelligent communication in human-computer interaction [28].

Finally, we have implemented RMM in three examples of multi-agent domains. Our aim has been to assess the reasonableness of the behavior resulting from our approach in a number of circumstances, and to assess its robustness and performance in mixed environments composed of RMM and human-controlled agents. Our experiments in mixed environments are intended to show the advantage of RMM as a mechanism for coordination that relies on modeling the other agents' rationality, as opposed to relying on coordination protocols. We briefly describe our results below.

We should note that all of the examples of coordination below were achieved without any communication among the RMM-based and the human-controlled agents that participated. The results of interactions with communication are reported in a separate paper.

5.1 Coordination in the Air Defense Domain

Our air defense domain consists of some number of anti-air units whose mission is to defend a specified territory from a number of attacking missiles (see Figure 5). The defense units have to coordinate and decide which missiles to intercept, given the characteristics of the threat, and given what they can expect of the other defense units. The utility of the agents' actions in this case expresses the desirability of minimizing the damage to the defended territory. The threat of an attacking missile was assessed based on the size of its warhead and its distance from the defended territory. Further, the defense units considered the hit probability, $P(H)$, with which their interceptors would be effective against each of the hostile missiles. The product of this probability and a missile threat was the measure of the expected utility of attempting to intercept the missile.

In this domain, it is easy to see the advantage of using decision-theoretic approach to game theory as implemented in RMM vs. the traditional game-theoretic solution concept of equilibria. Apart from the need for common knowledge the agents have to share to justify equilibria (we discuss this further in Section 6), the problem is that there may be many equilibria and no clear way to choose the "right" one to guide the agent's behavior. Take an example of two air defense units facing an attack by two missiles, A and B. In their choice of which missile the agents should intercept there are already two equilibria: One in which first agent intercepts A and second agent intercepts B, and a another one in which first agent intercepts B and second agent intercepts A. With the number of equilibria equal, or sometimes greater, than the number of alternative targets the agent would be left with no guidance as to which solution should be acted upon, and which threat should actually be intercepted next.

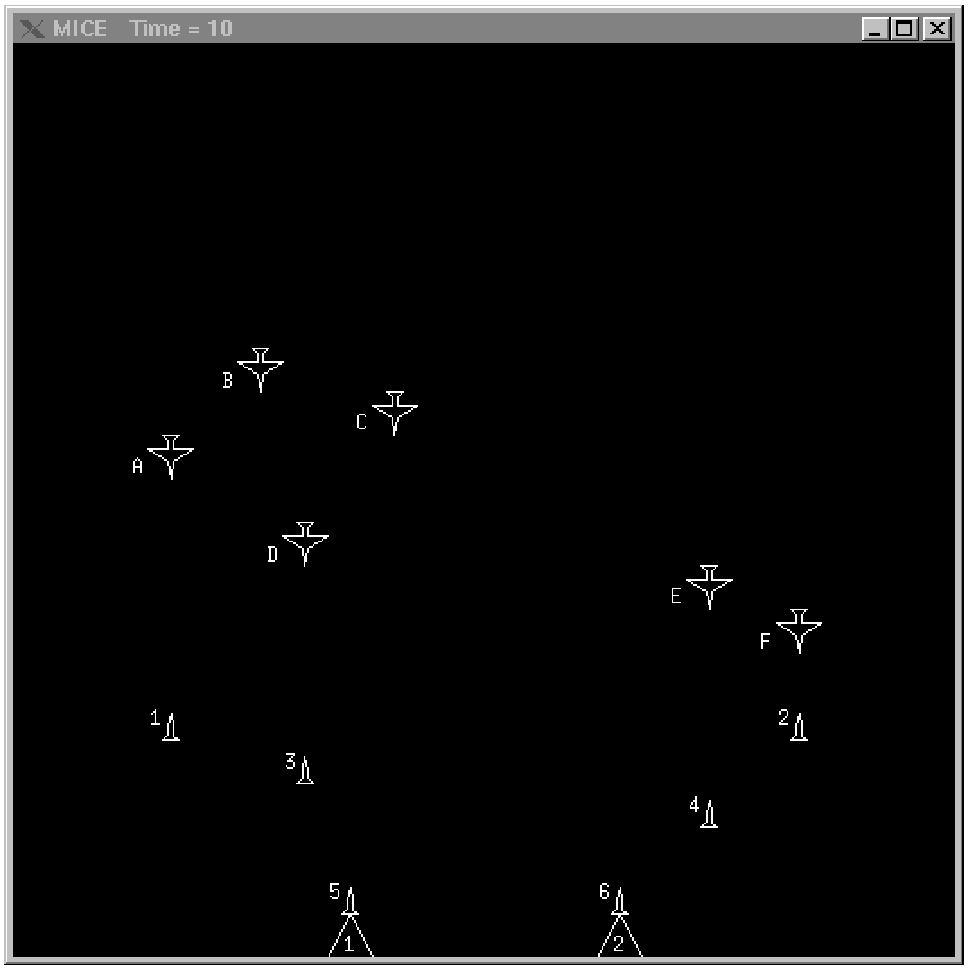

Figure 5: MICE Simulation of the Air Defense Domain.

In all of the experiments we ran[16], each of two defense units could launch three interceptors, and were faced with an attack by six incoming missiles. We put all of the experimental runs under the following conditions. First, the initial positions of missiles were randomly generated and it was assumed that each missile must occupy a distinct position. Second, the warhead sizes were $470, 410, 350, 370, 420, 450$ for missiles A through F, respectively. Third, the other battery was assumed to be operational with probability 0.8, and to be incapacitated with probability 0.2. Fourth, the performance assessments of agents with different policies were compared using the same threat situation. Further, each interceptor could intercept only one missile and it was moving twice as fast as the incoming missile. Finally, although there was no communication between agents, each agent could see which threat was shot at by the other agent and use this information to make its next decision.

Our experiments was aimed at determining the quality of modeling and coordination

[16]For an on-line demonstration of the air defense domain refer to the Web page http://dali.uta.edu/Air.html.

achieved by the RMM agents in a team, when paired with human agents, and when compared to other strategies. To evaluate the quality of the agents' performance, the results were expressed in terms of (1) the number of intercepted targets, i.e., targets the defense units attempted to intercept, and (2) the total expected damage to friendly forces after all six interceptors were launched. The total expected damage is defined as a sum of the residual warhead sizes of the attacking missiles. Thus, if a missile was targeted for interception, then it contributed $(1 - P(H)) \times Size$ to the total damage. If a missile was not targeted, it contributed all of its warhead size to the expected damage.

The target selection strategies are as follows:

- Random: selection randomly generated.
- Independent, no modeling: selection of $\arg\max_j\{P(H_{ij}) \times T_j\}$ for agent i.
- Human:[17] selection by human.
- RMM: selection by RMM.

The random agents were included to provide the worst-case base line of the system performance in our experiments. We included the independent agents to show what coordination can be expected when agents maximize but do not model each other in making their choices. We experimented with the above policies to understand the agent interactions in two groups: heterogeneous teams of agents with the same policy and the mixed agent teams with different policies.

As shown in Figure 6 and Figure 7, we found that the all-RMM team outperformed the human and independent teams. The average number of intercepted targets by the all-RMM team during 100 trials was 5.49 ($\sigma_{\bar{X}}$[18]$= 0.05$), compared to 4.89 ($\sigma_{\bar{X}} = 0.08$) for the independent team and 4.77 ($\sigma_{\bar{X}} = 0.06$) for the all-human team. Further, the RMM-controlled coordinated defense resulted in the total expected damage of 488.0 ($\sigma_{\bar{X}} = 23.4$), which was much less than that of the independent team ($732.0, \sigma_{\bar{X}} = 37.1$) and that of the all-human team ($772.0, \sigma_{\bar{X}} = 36.3$).

We found that the human performance was very similar to the performance of independent agents. The most obvious reason for this is that humans tend to depend on their intuitive strategies for coordination, and, in this case, found it hard to engage in deeper, normative, decision-theoretic reasoning. Sometimes the ways human subjects choose a missile were different and quite arbitrary. Some of them attempted to intercept the 3 left-most or right-most missiles, depending whether they were in charge of the left or the right defense battery. This led to difficulties when the missiles were clustered at the center area and to much duplicated effort. Others tended to choose missiles with the largest missile size. Still others tried to

[17] We should remark that our human subjects were CSE and EE graduate students who were informed about the criteria for target selection. We would expect that anti-air specialists, equipped with a modern defense doctrine, could perform better than our subjects. However, the defense doctrine remains classified and was not available to us at this point.

[18] $\sigma_{\bar{X}}$ denotes the standard error of the mean.

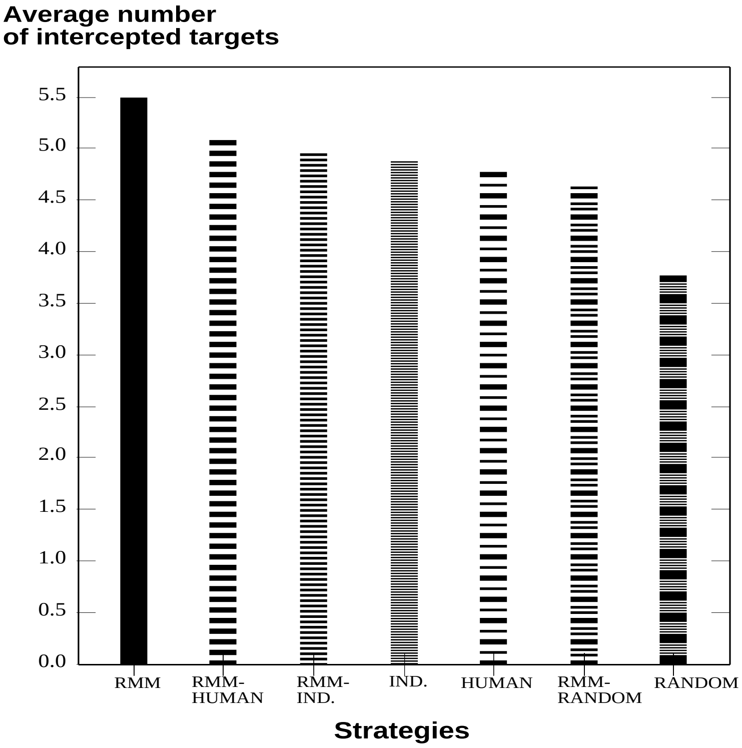

Figure 6: Average number of intercepted targets (over 100 runs).

consider the multiplication of the missile size and the hit probability, but did not model the other agent appropriately. The performance of the RMM team was not perfect, however, since the agents were equipped with limited and uncertain knowledge of each other.

The performance of the heterogeneous teams again suggests the favorable quality of coordination achieved by RMM agents. Comparing a heterogeneous team with a homogeneous team, the average number of intercepted targets for the RMM-Human team was 5.10 ($\sigma_{\bar{X}} = 0.04$), and for the all-human team 4.77; 4.98 ($\sigma_{\bar{X}} = 0.03$) for the RMM-Independent team vs. 4.89 for the Independent team; 4.66 ($\sigma_{\bar{X}} = 0.06$) for the RMM-Random team is, and 3.77 for the all-Random team.

In order to test whether the observed differences among the target selection strategies were not due to chance, we used an analysis of variance with a 0.01 significance level. Here, the all-human team and the RMM-Human team were left out, because of the relatively small number of participating human subjects.[19] In the experiment in which the number of intercepted targets was measured (Figure 6), $F(4, \infty) = 4.12$, $p < 0.01$. Therefore, we can conclude that the differences among the five target selection strategies are not due to chance with the probability 99%. This result holds also for the experiment in which the total expected damage was measured. To test the significance of the observed superiority of coordination achieved

[19]In our preliminary experiment there were 4 pairs of all-human teams and RMM-Human teams.

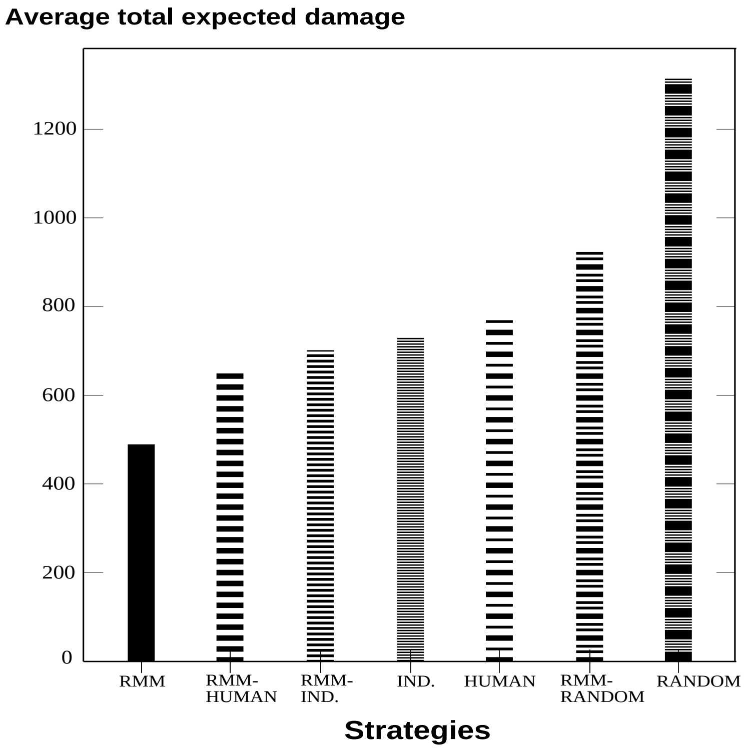

Figure 7: Average total expected damage (over 100 runs).

by the RMM team vs. the other non-human teams, t tests were performed. The results show that the RMM team was better than any other team with the probability of 99% (0.01 level of significance).

The above results show that RMM allows the automated agents to achieve high quality coordination in this unpredictable environment without relying on predefined protocols. As we argued, methods using traditional game-theoretic equilibria would not be sufficient to coordinate agents in our domain. A particularly promising facet of our results is that the Recursive Modeling Method is a robust mechanism for modeling and coordination not only among RMM agents, but also with the human-controlled agents.

5.2 Coordination in the Pursuit Problem

The pursuit problem is usually described as one during which four agents, called predators, have to coordinate their movements to pursue, surround, and capture the fifth agent, called prey (see Figure 8). Our RMM implementation of the predators' decision-making uses the evaluation of expected utility of alternative positions of the agents, resulting from their alternative moves, including the factors of how close the agents are to the prey, and how well the prey is surrounded and blocked off, as discussed in [49]. The expected utilities of alternative moves were then assembled into payoff matrices and used by the RMM agents in recursive model structures that are assumed to end on the fifth level with no-information models.

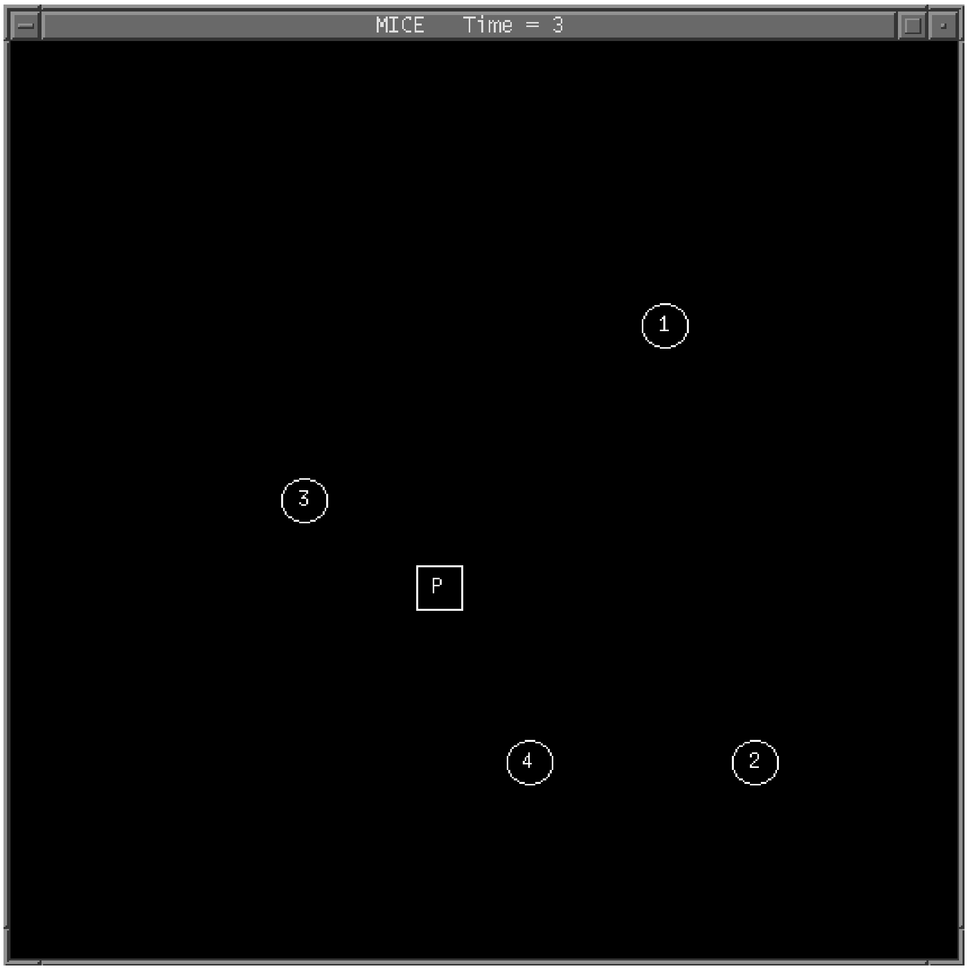

Figure 8: Predator-prey Coordination Game Simulated in MICE

In this domain we ran five sets of experiments, each consisting of five runs initialized by a randomly generated configuration of predators and prey. The five sets of runs contained different numbers of RMM and human agents, and typical runs in each set can be viewed at http://dali.uta.edu/Pursuit.html.

Using the time-to-capture as the measure of quality of the coordination among predators, we found that the best results were obtained by the all-human team (average time-to-capture of about 16 time units), followed by the all RMM team (average time-to-capture about 22 units), with the mixed RMM-human teams exhibiting the times of about 24 time steps (typical standard deviation for a set was 3.8). However, a statistical significance test (ANOVA) shows that the differences in the results obtained were not statistically significant; the difference in performance was not due to chance with probability less than 0.95.

Thus, the RMM-controlled agents were fairly competent in coordinating, but did not perform as well as human subjects. We think that the high quality results obtained by humans can be explained by the highly visual character of the task. Humans made their choices by eyeing the screen and choosing their actions based on how best to surround the prey. RMM agents, of course, did not have the advantage of visual input.

5.3 Cooperative Assembly Domain

We simulated a cooperative assembly task, characteristic of many space and manufacturing applications, using the blocks world in which the agents were to assemble the blocks into simple given configurations. In this domain, again, we tested the behavior of RMM agents when paired off with other RMM and human agents. The point was to observe the agents properly dividing the tasks of picking up various blocks, and not wasting the effort in attempting to pick up the same blocks. We have not performed rigorous analysis of performance achieved by agents in this domain but the reader can find the typical runs on http://dali.uta.edu/Blocks.html.

In summary, our experiments in the three domains above provide a promising confirmation of the ability of the RMM algorithm to achieve coordination among agents in unstructured environments with no pre-established coordination protocols. We found the behavior of RMM agents to be reasonable and intuitive, given that there was no possibility of communication. RMM agents were usually able to predict the behavior of the other agents, and to successfully coordinate with them. Given the nature of the application domains we outlined earlier and the frequent need for competence in interactions with humans, we find the experiments involving a heterogeneous mix of RMM and human participants particularly promising.

6 Related Work

Some of the most relevant works are ones that bear upon our Assumption 1 in Section 3.2, postulating finiteness of knowledge nesting in the recursive model structure.[20] A well-known particular case of infinitely nested knowledge is based on the notion of *common knowledge* [2]. A proposition, say p, is common knowledge if and only if everyone knows p, and everyone knows that everyone knows p, and everyone knows that everyone knows that everyone knows p, and so on *ad infinitum*. However, in their well-known paper [34], Halpern and Moses show that, in situations in which agents use realistic communication channels which can lose messages or which have uncertain transmission times,[21] common knowledge is not achievable in finite time unless agents are willing to "jump to conclusions," and assume that they know more than they really do.[22]

[20] Here, knowledge about the world is taken as something the agent is acquiring through sensing, as opposed to merely assuming.

[21] To our best knowledge, all practically available means of communication have such imperfections.

[22] Halpern and Moses consider the concepts of epsilon common knowledge and eventual common knowledge. However, in order for a fact to be epsilon or eventual common knowledge, other facts have to be common knowledge within the, so called, view interpretation. See [34] for details. Also, it has been argued that common knowledge can arise due to the agents' copresence, and, say, visual contact. These arguments are intuitive,

In other related work in game theory, researchers have begun to investigate the assumptions and limitations of the classical equilibrium concept [5, 26, 41, 64, 76]. As we mentioned, our work on RMM follows an alternative approach, proposed in [3, 7, 41, 62], and called a decision-theoretic approach to game theory. Unlike the outside observer's point of view in classical equilibrium analysis, the decision-theoretic approach takes the perspective of the individual interacting agent, with its current subjective state of belief, and coincides with the subjective interpretation of probability theory used in much of AI (see [12, 55, 59] and the references therein). Its distinguishing feature seems best summarized by Myerson ([53], Section 3.6):

> The decision-analytic approach to player i's decision problem is to try to predict the behavior of the players other than i first, and then to solve i's decision problem last. In contrast, the usual game-theoretic approach is to analyze and solve the decision problems of all players together, like a system of simultaneous equations in several unknowns.

Binmore [5] and Brandenburger [7] both point out that unjustifiability of common knowledge leads directly to the situation in which one has to explicitly model the decision-making of the agents involved given their state of knowledge, which is exactly our approach in RMM. This modeling is not needed if one wants to talk only of the possible equilibria. Further, Binmore points out that the common treatment in game theory of equilibria without any reference to the equilibrating process that achieved the equilibrium[23] accounts for the inability of predicting which particular equilibrium is the right one and will actually be realized, if there happens to be more than one candidate.[24]

The definition of the recursive model structure we presented is also closely related to interactive belief systems considered in game theory [3, 37, 52]. Our structures are somewhat more expressive, since they also include the sub-intentional and no-information models. Thus, they are able to express a richer spectrum of the agents' decision making situations, including their payoff functions, abilities, and information they have about the world, but also the possibility that other agents should be viewed not as intentional utility maximizers, but as mechanisms or simple objects.

Apart from game theory we should mention related work in artificial intelligence. In his philosophical investigations into the nature of intentions Bratman [8] distinguishes between mere plans, say as behavioral alternatives, and mental states of agents when they "have a plan in mind" which is relevant for having an intention (see also [1]). Our approach of viewing intentions as the results of rational deliberations over alternatives for action, given an agent's beliefs and preferences, is clearly very similar. Closely related is also the concept of practical rationality in [61]. Another strand of philosophical work that we follow, as we have mentioned

but turn out to be difficult to formalize, so we treat the issue here as open.

[23]Binmore compares it to trying to decide which of the roots of the quadratic equation is the "right" solution without reference to the context in which the quadratic equation has arisen.

[24]Binmore [6], as well as others in game theory [42, 43, 14, 15] and related fields [72], suggest the evolutionary approach to the equilibrating process. The centerpiece of these techniques lies in methods of belief revision, which we also investigated using the RMM framework. Some of our results are presented in [31, 74].

before, is Dennett's formulation of the intentional stance [19], and his idea of the ladder of agenthood (see [54] for a succinct discussion), the first five levels of which we see as actually embodied in RMM. Somewhat related to RMM is the familiar minimax method for searching game trees [57]. However, game tree search assumes turn taking on the part of the players during the course of the game and it bottoms out when the game terminates or at some chosen level, while RMM addresses agent's choice without observing the other agents' moves and it bottoms out when there is no more knowledge.

Shoham's agent-oriented programming (AOP) [71] takes more of a programming-language perspective. Shoham defines many mental attitudes, for example belief, obligation, and choice, as well as many types of messages that the agents can exchange, and he has developed a preliminary version of an interpreter. However, while Shoham has proposed it as an extension, decision-theoretic rationality has not yet been included in AOP.

The issue of nested knowledge has also been investigated in the area of distributed systems [22] (see also [21]). In [22] Fagin and colleagues present an extensive model-theoretic treatment of nested knowledge which includes a no-information extension (like the no-information model in RMM) to handle the situation where an agent runs out of knowledge at a finite level of nesting; however, no sub-intentional modeling is envisioned. Further, they do not elaborate on any decision mechanism that could use their representation (presumably relying on centrally designed protocols). Another related work on nested belief with an extensive formalism is one by Ballim and Wilkes [4]. While it concentrates on mechanisms for belief ascription and revision, primarily in the context of communication, it does not address the issues of decision making. Korf's work on multi-agent decision trees considers issues of nested beliefs, where the beliefs that agents have about how each evaluates game situations can vary [46]. Tambe describes another interesting approach to coordinating agents during team activities in [75].

The applications of game-theoretic techniques to the problem of interactions in multi-agent domains have also received attention in the Distributed AI literature, for example in [65, 66, 67]. This work uses the traditional game-theoretic concept of equilibrium to develop a family of rules of interaction, or protocols, that would guarantee the properties of the system as a whole that are desirable by the designer, like stability, fairness and global efficiency. Other work by Koller [44] on games with imperfect information, Wellman's WALRAS system [80, 79], and Sandholm work on coalitions [70] also follow the more traditional lines of equilibrium analysis.

7 Complexity

One look at the branching nested representations proposed in this paper is enough to suggest that complexity may become an issue. Indeed, if we were to characterize the size of a problem for RMM to solve by the number of agents, n, it is easy to show that the complexity of building and solving the recursive models grows exponentially as $O(|A|^n * m^l)$, where $|A|$ is the number of alternative actions considered, m is the branching factor of the recursive model structure, and l is the level of nesting of the model structure.

Luckily, an exhaustive evaluation of the full-blown RMM hierarchy can be simplified in a number of ways. For lack of space, we briefly list some of the most intuitive methods (see [27] and the more recent [58] for more details). First, the dynamic programming solution of the recursive model structure takes advantage of the property of *overlapping subproblems* (see [18], section 16.2), which avoids repeated redundant solutions of branches with the same form in the recursive model structure. The extent to which problems do overlap, is, of course, case dependent. However, in environments like the pursuit problem, described in Section 5, the overlap in subproblems leads to reducing complexity down to a polynomial.

A powerful idea for further reducing the complexity of agent coordination in large groups is to neglect the models of agents with which the interaction is weak. First, it can be shown that models of some agents can be safely neglected, since they possibly cannot change the solution for the best alternative. Second, some models that potentially could influence the solution will do so with only a very small probability. This family of simplifications and approximations is clearly similar to strategies of coordinating humans; we usually worry about the people in our immediate vicinity and about the few persons we interact with most closely, and simply neglect the others within, for example, the building, organization, or the society at large. As it turns out, the payoff matrices lend themselves to an efficient assessment of the strength of interaction between agents by analyzing variability of the payoff values. For details of these and other simplification methods, see [27, 58, 77], and related work in [60, 69].

8 Summary and Conclusions

The starting point for our explorations in this paper has been the presumption that coordination should emerge as a result of rational decisions in multi-agent situations, where we defined rationality as maximization of expected utility. We argued that decision-theoretic rationality is applicable to multi-agent interactions since the agents have to make choices under uncertainty: The abilities, sensing capabilities, beliefs, goals, preferences, and intentions of other agents are not directly observable and usually are not known with certainty. Thus, we used decision-theoretic rationality as a normative paradigm, describing how an agent should make decisions in an uncertain multi-agent environment.

Further, we used expected utility maximization as a descriptive paradigm, to model other rational agents in a multi-agent environment. We have documented how our exploration naturally brings us to concepts from game theory, but our concern with providing a decision-making apparatus to an individual agent, rather than providing an observer with analytical tools, has led us away from the traditional concern with equilibrium solutions. Instead, we use a newly proposed decision-theoretic approach to game theory, implemented using dynamic programming. Our agent-centered perspective, as well as our assumption that the knowledge of the agent is finitely nested, are the two main differences between our approach in RMM and the traditional game theoretic analysis. We argued that the solution concept presented in this paper complements the game-theoretic solution: When the knowledge of an agent is nested down to a finite level, a decision-theoretic approach implemented using dynamic programming is applicable. When the infinitely nested common knowledge is available the bottom-up dynamic programming is not applicable, but a fixed-point solution, corresponding to the classical Nash equilibrium, can be sought using traditional game-theoretic methods.

Our implementation of the RMM algorithm in three multi-agent domains supports our claim that coordination emerges as a result of rational decision-making of agents interacting when they have no pre-established protocols to guide them. We found that the RMM agents are able to coordinate on the level comparable to that of the human-controlled agents, and, in some cases, better. Particularly encouraging is the ability of the RMM agents to effectively interact not only with other RMM agents, but also with other agents and humans.

Our investigations can be extended in numerous ways. First, in practical situations, the intentional stance can be only one of the guides to the expected behavior of other agents; the agents also have to be able to update models of each other through observation and plan recognition. The challenge is in integrating the normative, intentional modeling using other's rationality with techniques based on observation. Our work in this direction utilizes Bayesian learning, for which RMM, given its probabilistic character, is naturally suited (see [31, 74] for recent results). Second, we are exploring how the deeper reasoning in RMM, having been done once, can be summarized (compiled) into shallower, models of other agents or heuristic rules of interactive behavior. This means that, even in cases where an agent cannot afford to use RMM in deciding what it should do in a time-constrained situation and resorts to a (possibly wrong) heuristic response, an agent can revisit previous decision situations when it has the time and use RMM to determine what the rational response *should* have been. By storing this as a rule of behavior that can be recalled when appropriate in the future (see related work on chunking [47]), RMM can provide the basis for the accrual of rational heuristics [58].

Another important direction, and an application area, of RMM is studying rational communicative behavior among agents involved in interactions. It turns out that our framework allows the agents to also compute the expected utilities of alternative communicative actions they could execute. The agent's maximizing the expected utility of such actions leads to rational communicative behavior. We will report on our approach, implementation and results in this area in a forthcoming paper.

Acknowledgments

We would like to gratefully acknowledge the helpful comments and encouragement we received from our colleagues in the distributed AI community, as well as from people outside the distributed AI community who graciously provided us with further insights and/or sanity checks. In particular, we would like to thank Professor David Wehe from the Department of Nuclear Engineering at the University of Michigan, Professor Eddie Dekel from the Department of Economics at Northwestern University, Professor Cristina Bicchieri from the Department of Philosophy at Carnegie Mellon University, Professor Adam Brandenburer from the Harvard Business School, and Professor Hal Varian from the Department of Economics at the University of California at Berkeley.

We also acknowledge the help of our students, Jose Vidal from the University of Michigan's EECS department, and Tad Kellogg and Sanguk Noh from the University of Texas at

Arlington's CSE department, for their invaluable help in implementing and experimenting with RMM.

References

[1] James F. Allen. Two views of intention. In P. R. Cohen, J. Morgan, and M. E. Pollack, editors, *Intentions in Communication*. MIT Press, 1990.

[2] Robert J. Aumann. Agreeing to disagree. *Annals of Statistics*, 4(6):1236–1239, 1976.

[3] Robert J. Aumann and Adam Brandenburger. Epistemic conditions for Nash equilibrium. *Econometrica*, 1995.

[4] Afzal Ballim and Yoric Wilks. *Artificial Believers*. Earlbaum Associates, Inc., 1991.

[5] Ken Binmore. *Essays on Foundations of Game Theory*. Pitman, 1982.

[6] Ken Binmore. Rationality in the centipede. In *Proceedings of the Conference on Theoretical Aspects of Reasoning about Knowladge*, pages 150–159. Morgan Kaufman, March 1994.

[7] Adam Brandenburger. Knowledge and equilibrium in games. *Journal of Economic Perspectives*, 6:83–101, 1992.

[8] Michael E. Bratman. What is intention? In P. R. Cohen, J. Morgan, and M. E. Pollack, editors, *Intentions in Communication*. MIT Press, 1990.

[9] J. R. Busemeyer and J. T. Townsend. Decision field theory: A dynamic cognitive approach to decision making in an uncertain environment. *Psychological Review*, pages 432–457, 1993.

[10] Colin Camerer. Individual decision making. In John H. Kagel and Alvin E. Roth, editors, *The Handbook of Experimental Economics*. Princeton University Press, 1995.

[11] David Carmel and Shaul Markovitch. Learning models of intelligent agents. In *Proceedings of the National Conference on Artificial Intelligence*, pages 62–67, Portland, OR, August 1996.

[12] Peter Cheeseman. In defense of probability. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 1002–1009, Los Angeles, California, August 1985.

[13] H. Chernoff and L. E. Moses. *Elementary Decision Theory*. John Wiley, New York, 1959.

[14] In-Koo Choo and Akihiko Matsui. Induction and bounded rationality in repeated games. Technical report, CARESS Working Paper 92-16, University of Pennsylvania, May 1992.

[15] In-Koo Choo and Akihiko Matsuri. Learning and the ramsey policy. Technical report, CARESS Working Paper 92-18, University of Pennsylvania, June 1992.

[16] Ronald Christensen. *Multivariate Statistical Modeling.* Entropy Limited, 1983.

[17] L. S. Coles, A. M. Robb, P. L. Sinclar, M. H. Smith, and R. R. Sobek. Decision analysis for an experimental robot with unreliable sensors. In *Proceedings of the Fourth International Joint Conference on Artificial Intelligence*, Stanford, California, August 1975.

[18] Thomas H. Cormen, Charles E. Leiserson, and Ronald L. Rivest. *Introduction to Algorithms.* The MIT Press, 1990.

[19] D. Dennett. Intentional systems. In D. Dennett, editor, *Brainstorms.* MIT Press, 1986.

[20] Jon Doyle. Rationality and its role in reasoning. *Computational Intelligence*, 8:376–409, 1992.

[21] Ronald R. Fagin, Joseph Y. Halpern, Yoram Moses, and Moshe Y. Vardi. *Reasoning About Knowledge.* MIT Press, 1995.

[22] Ronald R. Fagin, Joseph Y. Halpern, and Moshe Y. Vardi. A model-theoretic analysis of knowledge. *Journal of the ACM*, (2):382–428, April 1991.

[23] Jerome A. Feldman and Robert F. Sproull. Decision theory and Artificial Intelligence II: The hungry monkey. *Cognitive Science*, 1(2):158–192, April 1977.

[24] Peter Fishburn. *The Foundations of Expected Utility.* D. Reidel Publishing Company, 1981.

[25] K. Forbus. Spatial and qualitative aspects of reasoning about motion. In *Proceedings of the First Annual National Conference on Artificial Intelligence*, Stanford, California, July 1980.

[26] John Geanakoplos. Common knowledge. In *Proceedings of the Conference on Theoretical Aspects of Reasoning about Knowladge*, pages 255–315. Morgan Kaufman, August 1992.

[27] Piotr Gmytrasiewicz. *A Decision-Theoretic Model of Coordination and Communication in Autonomous Systems.* PhD thesis, University of Michigan, December 1991.

[28] Piotr J. Gmytrasiewicz. An approach to user modeling in decision support systems. In *Proceedings of the Fifth International Conference on User Modeling*, pages 121–127, January 1996.

[29] Piotr J. Gmytrasiewicz, Edmund H. Durfee, and David K. Wehe. Combining decision theory and hierarchical planning for a time-dependent robotic application. In *Proceedings of the Seventh IEEE Conference on AI Applications*, pages 282–288, February 1991.

[30] Piotr J. Gmytrasiewicz, H. H. Huang, and Frank L. Lewis. Combining operations research and agent-oriented techniques for agile manufacturing system design. In *The Proceedings of the IASTED International Conference on Robotics and Manufacturing*, June 1995.

[31] Piotr J. Gmytrasiewicz, Sanguk Noh, and Tad Kellogg. Bayesian update of recursive agent models. *International Journal of User Modeling and User-Adapted Interaction, Special Issue on Machine Learning for User Modeling*, 1998. Available in postscript from http://www-cse.uta.edu/ piotr/piotr.html.

[32] Peter Haddawy and Steven Hanks. Issues in decision-theoretic planning: Symbolic goals and numeric utilities. In *Proceedings of the 1990 DARPA Workshop on Innovative Approaches to Planning, Scheduling, and Control*, pages 48–58, November 1990.

[33] Joseph Y. Halpern. Reasoning about only knowing with many agents. In *Proceedings of the National Conference on Artificial Intelligence*, pages 655–661, July 1993.

[34] Joseph Y. Halpern and Yoram Moses. Knowledge and common knowledge in a distributed environment. *Journal of the ACM*, 37(3):549–587, July 1990.

[35] Walter C. Hamscher. Modeling digital circuits for troubleshooting. *Artificial Intelligence*, 51:223–271, 1991.

[36] Steven Hanks and R. James Firby. Issues in architectures for planning and execution. In *Proceedings of the Workshop on Innovative Approaches to Planning, Scheduling and Control*, pages 59–70, November 1990.

[37] John C. Harsanyi. Games with incomplete information played by 'Bayesian' players. *Management Science*, 14(3):159–182, November 1967.

[38] David Heckerman and David Maxwell Chickering. A comparison of scientific and engineering criteria for bayesian model selection. Technical Report MSR-TR-96-12, Microsoft Research, Microsoft Corporation, Redmond, WA, 1996.

[39] Marcus J. Huber, Edmund H. Durfee, and Michael P. Wellman. The automated mapping of plans for plan recognition. In *Proceedings of 1994 Conference on Uncertainty in Artificial Intelligence*, pages 344–351, August 1994.

[40] W. Jacobs and M. Kiefer. Robot decisions based on maximizing utility. In *Proceedings of the Third International Joint Conference on Artificial Intelligence*, pages 402–411, August 1973.

[41] Joseph B. Kadane and Patrick D. Larkey. Subjective probability and the theory of games. *Management Science*, 28(2):113–120, February 1982.

[42] Michihiro Kandori, George J. Mailath, and Rafael Rob. Learning, mutation and long run equilibria in games. Technical report, CARESS Working Paper 91-01R, University of Pennsylvania, January 1991.

[43] Michihiro Kandori and Rafael Rob. Evolution of equilibria in the long run: A general theory and applications. Technical report, CARESS Working Paper 91-01R, University of Pennsylvania, January 1991.

[44] Daphne Koller and Avi Pfeffer. Generating and solving imperfect information games. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, pages 1185–1192, Montreal, Canada, August 1995.

[45] Daphne Koller and Avi Pfeffer. Probabilisitc frame based systems. In *Proceedings of the National Conference on Artificial Intelligence*, pages 580–587, Madison, WI, July 1998.

[46] Richard E. Korf. Multi-agent decision trees. In *Proceedings of the 1989 Distributed AI Workshop*, pages 293–306, September 1989.

[47] John Laird, Paul Rosenbloom, and Allen Newell. *Universal Subgoaling and Chunking: The automatic generation and learning of goal hierarchies.* Kluwer Academic Publishers, 1986.

[48] Gerhard Lakemeyer. All they know about. In *Proceedings of the National Conference on Artificial Intelligence*, July 1993.

[49] Ron Levy and Jeffrey S. Rosenschein. A game theoretic approach to the pursuit problem. In *Working Papers of the Eleventh International Workshop on Distributed Artificial Intelligence*, pages 195–213, February 1992.

[50] Per-Eric Malmnas. Axiomatic justification of the utility principle: A formal investigation. *Synthese*, (99):233–249, 1994.

[51] J. E. Mardsen and A. J. Tromba. *Vector Calculus.* W. H. Freeman and Company, 1981.

[52] Mertens and Zamir. Formulation of Bayesian analysis for games with incomplete information. *International Journal of Game Theory*, 14:1–29, 1985.

[53] Roger B. Myerson. *Game Theory: Analysis of Conflict.* Harvard University Press, 1991.

[54] Ajit Narayanan. *On Being a Machine.* Ellis Horwood, 1988.

[55] Richard E. Neapolitan. *Probabilistic Reasoning in Expert Systems.* John Wiley and Sons, 1990.

[56] Allen Newell. The knowledge level. *AI Magazine*, 2(2):1–20, Summer 1981.

[57] Nils J. Nilsson. *Problem-Solving Methods in Artificial Intelligence.* McGraw-Hill, 1971.

[58] Sanguk Noh and Piotr J. Gmytrasiewicz. Towards flexible multi-agent decision-making under time pressure. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, pages 492–498, August 1999.

[59] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufman, 1988.

[60] Kim L. Poh and Eric J. Horvitz. Reasoning about the value of decision-model refinement: Methods and application. In *Proceedings of the Ninth Conference on Uncertainty on Uncertainty in Artificial Intelligence (UAI-93)*, August 1993.

[61] J. L. Pollock. *How to Build a Person: A Prolegomenon.* MIT Press, 1989.

[62] H. Raiffa. *The Art and Science of Negotiation.* Harvard University Press, 1982.

[63] Eric Rasmusen. *Games and Information.* B. Blackwel, 1989.

[64] Philip J. Reny. Extensive games and common knowledge. In *Proceedings of the Conference on Theoretical Aspects of Reasoning about Knowladge*, page 395. Morgan Kaufman, 1988.

[65] Jeffrey S. Rosenschein and John S. Breese. Communication-free interactions among rational agents: A probablistic approach. In Les Gasser and Michael N. Huhns, editors, *Distributed Artificial Intelligence*, volume 2 of *Research Notes in Artificial Intelligence*, pages 99–118. Pitman, 1989.

[66] 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. (Also published in *Readings in Distributed Artificial Intelligence*, Alan H. Bond and Les Gasser, editors, pages 227–234, Morgan Kaufmann, 1988.).

[67] Jeffrey S. Rosenschein and Gilad Zlotkin. *Rules of Encounter.* MIT Press, 1994.

[68] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach.* Prentice Hall, 1995.

[69] S. Russell and E. Wefald. On optimal game tree search using rational meta-reasoning. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 334–340, Detroit, Michigan, August 1989.

[70] Tuomas Sandholm. Coalitions among computationally bounded agents. *Artificial Intelligence, Special Issue on Economic Principles of Multiagent Systems*, 94(1):99–137, 1997.

[71] Yoav Shoham. Agent-oriented programming. *Artificial Intelligence*, 60(1):51–92, 1993.

[72] John Maynard Smith. *Evolution and the theory of games.* Cambrigde University Press, 1982.

[73] Reid G. Smith. The contract net protocol: High-level communication and control in a distributed problem solver. *IEEE Transactions on Computers*, C-29(12):1104–1113, December 1980.

[74] Dicky Suryadi and Piotr J. Gmytrasiewicz. Learning models of other agents using influence diagrams. In *Preceedings of the 1999 International Conference on User Modeling*, pages 223–232, Banf, CA, July 1999.

[75] Milind Tambe. Towards flexible teamwork. *Journal of Artificial Intelligence Research*, 7, 1997.

[76] Tommy C. Tan and Sergio R.D.C. Werlang. A guide to knowledge and games. In *Proceedings of the Second Conference on Theoretical Aspects of Reasoning about Knowledge*, 1988.

[77] Jose Vidal and Edmund Durfee. Recursive agent modeling using limited rationality. In *In Proceedings of the First International Conference on Multiagent Systems*, June 1995.

[78] Michael P. Wellman. The preferential semantics for goals. In *Proceedings of the National Conference on Artificial Intelligence*, pages 698–703, July 1991.

[79] Michael P. Wellman. A market-oriented programming environment and its applications to distributed multicommodity flow problems. *Journal of Artificial Intelligence Research*, 1:1–22, 1993.

[80] Michael P. Wellman. A computational market model for distributed configurational design. *AI EDAM*, 1995.