Coordination and Belief Update in a Distributed Anti-Air Environment*

Sanguk Noh and Piotr J. Gmytrasiewicz
Department of Computer Science and Engineering
University of Texas at Arlington
{noh, piotr}@cse.uta.edu

Abstract

This research addresses rational decision making and coordination among automated and human-controlled anti-air units whose mission is to defend a specified territory from a number of attacking missiles. The automated units have to decide which missiles to attempt to intercept, given the characteristics of the threat, and given the other units' anticipated actions, in their attempt to minimize the expected overall damages to the defended territory. Thus, an automated defense unit needs to model the other agents, either human or automated, that control the other defense batteries. In this paper, we assume that the units cannot communicate among themselves, say, due to an imposed radio silence. We use the Recursive Modeling Method (RMM), which enables an agent to select his rational action by examining the expected utility of his alternative behaviors, and to coordinate with other agents by modeling their decision making in a multiagent environment. Bayesian learning is used in conjunction with RMM for belief update. As a result, an agent can determine which models of the other agents are correct, and keep his knowledge up to date. We describe how decision making using RMM and Bayesian learning is applied to the anti-air defense domain and show experimental results.

1. Introduction

This paper describes rational decision making and rational coordination among the anti-air defense units facing a missile attack. The task of automated defense units is to defend a specified territory and to coordinate their attempts to intercept the attacking missiles, given the characteristics of the threat, and given what they can expect of the other defense units.

Under a realistic threat situation, friendly defense units cannot expect to have an advanced knowledge of the character of the incoming attack. It is, therefore, crucial that each of the defense units make a decision as to which incoming threat to intercept with an available interceptor by analyzing all potential threats acquired from the radar. In such cases, *coordination* requires an agent to recognize the current status, and to model the actions of the other agents to decide on his own next behavior. Since in any realistic combat situation the integrity of the defense team cannot be guaranteed, and even the very existence of the other friendly units cannot be counted on, relying on pre-established coordination protocols can be suboptimal or even dangerous. Our approach is to let each unit independently decide on and execute his action, and let coordination among the units emerge on the fly as the result of the units' individual rational actions.

We formulate the anti-air defense as a decision-theoretic problem from the point of view of an individual defense unit. The overall objective is to *minimize damages* to the defended territory. Since the classical notion of a symbolic goal doesn't provide a sufficient basis for choice of action that could minimize damages in uncertain situations [18], we need the attributes quantifying the quality of choices in the design of decision making procedures. First, each attacking missile has its threat value. We compute the missile's threat using such attributes as the altitude of the missile and the size of its warhead. Further, the defense units should consider the probability with which their interceptors would be effective against each of the hostile missiles. Based on these attributes combined, each unit has to determine the optimal action.

To fulfill the primary task successfully an automated defense unit needs to be able to anticipate the defense activities of other unit(s). To accomplish this, in absence of communication, the unit has to *model* the agents, either human or automated, controlling the other defense batteries. We adopt an approach to modeling other agents as rational utility maximizers [1] as

* This is an extended and revised version of a paper presented at the Workshop of Machine Learning for User Modeling [10]. This research has been sponsored by the Office of Naval Research Artificial Intelligence Program under contract N00014-95-1-0775, and by a research initiation grant from the CSE Department of the University of Texas at Arlington.

[1] This approach has been advocated and used extensively in the areas of modern economic analysis to, for example, model and predict consumer and customer behavior. However, it has begun to be widely used in AI applications only recently [3], even though formal notions of rationality

defined in the field of decision theory.

Our approach to modeling uses the Recursive Modeling Method (RMM), previously reported in [5]. RMM enables an agent to model the other agents and to rationally coordinate with them even if no protocol or overall plan can be established explicitly in advance. Using RMM as a decision making tool, an agent rationally selects his action under uncertainty guided by the principle of *expected utility maximization*. In our RMM framework, an agent uses the other agents' alternative models. Further, we use Bayesian learning, in conjunction with RMM, to *update the beliefs about the other agents based on their observed actions*. We feel that Bayesian belief update is particularly useful in dynamic environments characterized by unanticipated change, such as the anti-air defense.

We expect that coordination based on modeling will be appropriate to the anti-air defense domain, since, as we mentioned, the coordinating units have to be able to react to threats in previously unforeseen circumstances. These can include the changing nature of the attack, other defense units being shot at and possibly incapacitated, communication lines broken down, sudden need for complete radio silence, and so on. In these unexpected conditions, relying on a globally consistent view of the situation, achieved by communication or by pre-established protocols, is unrealistic or likely to lock the agents into suboptimal forms of behavior. Further, by being competent decision makers and able to rationally model the action of other agents, RMM agents can effectively interact with human-controlled units, in spite of a lack of a predetermined protocol that a human would have to learn and follow.

2. Anti-air defense environment

Let us consider a situation depicted in Figure 1. This scenario has six incoming missiles and two defending units in a 20 by 20 grid world. Each of the defense units independently decides to launch interceptors against the incoming missiles. The incoming missiles keep going straight down and attack the overall ground site on which the units are located.

2.1. Attribute analysis

One of the major problems encountered in connection with decision making process is how to model preferences and utilities. To formalize the decision making problem of minimizing damages, we first consider the attributes that influence the units' decision making. Each missile has its intrinsic threat value. In our model, threat is evaluated by considering the altitude

contributed to AI early on.

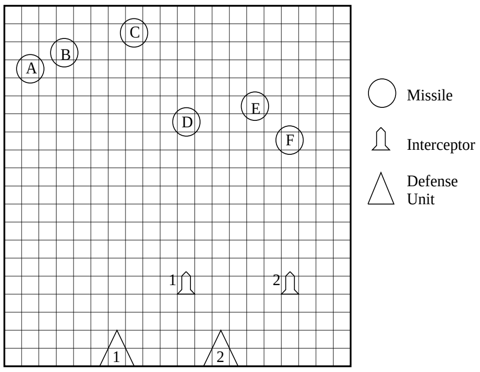


Figure 1: The anti-air defense scenario.

of a missile and its warhead size. Intuitively, a defense battery should give priority to the missile which is closer to the ground and bigger than the others. An explosion of a missile close to the ground results in a more powerful blast, which increases the damage. We calculate the missile threat, T, using the following formula:

$$T_n = W_n / A_n, \tag{1}$$

where
- W_n: the warhead size of missile n;
- A_n: the altitude of missile n.

A factor that does not enter into the missile threat calculation, but one that should be considered when an intercept decision is made, is the probability that an interceptor would be successful against a given missile. This probability is assumed to depend on the angle between a missile's direction of motion and the battery's line of sight. This probability is maximized when this angle is 0, as follows:

$$P(H_{ij}) = e^{-\mu \gamma_{ij}}, \tag{2}$$

where
- γ_{ij}: the angle between battery i's line of sight and missile j's direction of motion. Thus, $\gamma_{ij} = \tan^{-1}\alpha - \tan^{-1}\beta$ such that $0 \leq \gamma_{ij} \leq 90$;
- α: the slope of the missile j's direction of motion;
- β: the slope of the line of sight with which battery i aims at missile j;
- μ: an interceptor-specific constant (assumed here to be 0.01).

We will use the values of missile threat and the hit probability to develop the decision-theoretic assessment of the agent's alternative plans of defense action in an anti-air defense environment.

3. Decision-theoretic agent

Our decision-theoretic agent is based on the components depicted in Figure 2. In this design we have followed Shoham's agent oriented programming paradigm [16], according to which the mental state of agents can be represented in terms of their *beliefs*, *capabilities*, and *choices*. While in Shoham's work the behavior of an agent is guided by pre-defined rules, our agents rationally determine how to behave based on the principle of maximum expected utility (PMEU) [14]. PMEU can be implemented in the anti-air defense domain using the Recursive Modeling Method (RMM). RMM [5] will be used to model other agents and to coordinate with them. Further, Bayesian belief update will be applied to recognize the other agents' capabilities correctly.

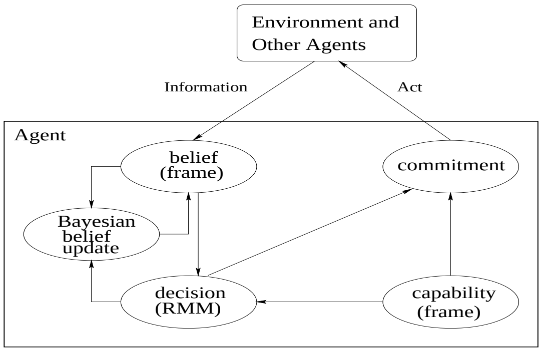
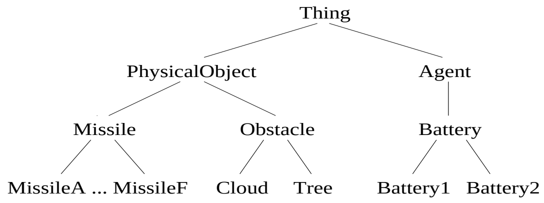


Figure 2: The components of agent's mental state.

To implement the components of an agent's mental state depicted in Figure 2 we use a version of the frame-based system [2]. Thus, frames are used to represent the agent's beliefs, capabilities, and preferences, and to organize his knowledge using the object-oriented paradigm into hierarchy of concepts and classes. Our rational agent processes his knowledge to determine optimal decisions, and to perform belief update to interactively maintain his knowledge base in a dynamic environment. In order to construct the well-defined frame-based system we adopt an ontological hierarchy of concepts. This $is-a$ hierarchy defines the existing object classes and their relations, as depicted below. The root class, Thing, consists of Agent and PhysicalObject subclasses. An agent is endowed with beliefs, capabilities, and choice. In contrast, a PhysicalObject has no mental state of its own and is assumed not to make any decisions. Missile and Obstacle are the subclasses of PhysicalObject and Battery is the subclass of Agent.



3.1. Generation of a payoff matrix

Our approach is to take the agent-oriented perspective. In scenario such as one in Figure 1, we view the decision making through the eyes of an individual defense unit, Battery1, and his radar-acquired data.[2] Figure 3 depicts the information acquired in the example scenario by Battery1 for the missiles A through F. In Figure 1, the left top corner of the screen is (0,0), x is pointing right, and y is pointing down. Applying formulas (1) and (2) to the acquired data, Battery1 can compute the relevant attributes of altitude, warhead size, and the angle γ that determines the hit probability. Battery1 also generates the expected hit probabilities for Battery2, assuming his hypothetical intercepting actions. The results summarized in Figure 3.

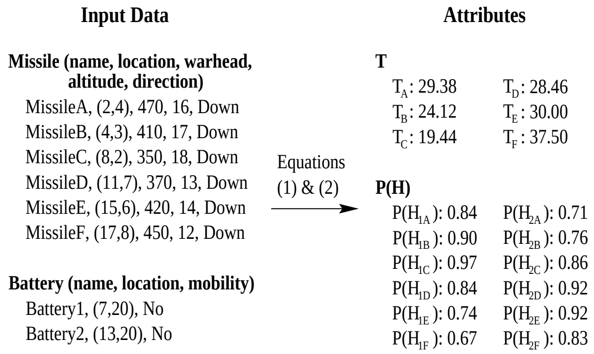


Figure 3: Radar data acquired by Battery1 for the missiles A through F and the derived attributes.

Commonly, a decision problem is represented by an influence diagram [6], which sometimes can be constructed dynamically. Poh and Horvitz [12] use influence diagrams that include decision variables, with values ranging over the possible decisions; chance variables, which represent the uncertainty of the domain; and a utility node. In our work, we rely on the payoff matrix representation used in game theory [9]. Payoff matrices, while different from influence diagrams, can be seen to faithfully summarize the information contained in them by listing the expected payoffs (obtained from the utility node) of possible decisions,

[2] Battery2 acquires the information about the environment from his point of view. Battery1 and Battery2 maintain their knowledge bases independently.

depending on the parameters describing the domain (chance nodes).

The expected payoffs, corresponding to batteries' attempting to intercept the respective missiles, can be expressed as a combination of the threat of the missiles and the probability of their interception. For example, if Battery1 is faced with n missiles at some state, and he targets a missile j, the resulting threat will be reduced by the missile threat T_j multiplied by the probability of successful interception $P(H_{1j})$. If both batteries target missiles at the same time, the reduction of threat, and therefore the total payoff, is equal to the sum of the threats that each of them removes.

4. Modeling and belief update

4.1. The recursive model structure

In order to solve his decision making problem, described by the payoff matrix generated as above, Battery1 needs to hypothesize the likely actions of Battery2. In the Recursive Modeling Method, the actions of the other rational agents are anticipated using a model of their decision making situation. We assume the following scenario: As shown in Figure 1, missiles A, B, C are coming toward Battery1 and missiles D, E, F toward Battery2. And the altitudes of missiles D, E, F are lower than those of missiles A, B, C. In this case, we allowed for uncertainty: Battery1 is uncertain whether Battery2 has any short range interceptors left, and whether he has been incapacitated by enemy fire. If Battery2 has only long range interceptors, he will be unable to attack any of the missiles D, E, or F, and can only attempt to shoot down one of A, B, or C.

From Battery1's point of view, therefore, Battery2's decision making situation could be modeled as one of three cases: Battery2 has both short and long range interceptors; Battery2 has only long range interceptors; or Battery2 has been damaged or incapacitated, in which case there is no information as to what action he would undertake. If Battery2 has only long range missiles he can only intercept the distant targets (A, B and C), even though he is faced with closer targets with larger threat value. Thus, there are three alternative models that Battery1 can use to model Battery2's decision making; one model has a fully developed payoff matrix with six alternative interception attempts corresponding to threats A through F, another has a payoff matrix with only three targets, and the third one contains no information about Battery2's action. We call the last model the *No-info* model. In RMM, each of the alternative models is assigned a probability indicating the likelihood of its correctness. Figure 4 is the Battery1's model structure of depth two for the example scenario.

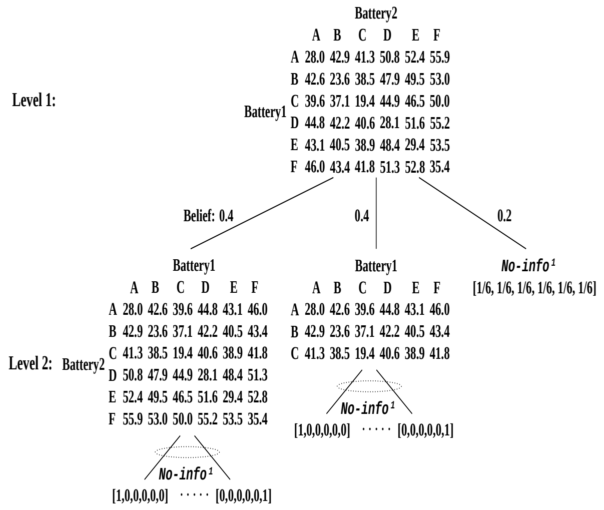


Figure 4: The recursive model structure for Battery1.

Thus, in the recursive model structure, Level 1 represents the way that Battery1 observes the situation to make his own decision, shown as Battery1's payoff matrix. Level 2 depicts the models Battery1 has of Battery2's situation. The recursive modeling could continue into deeper levels, but in this case we assumed that Battery1 has no further information. In other words, we are examining the reasoning of Battery1 in the particular case, when equipped with a finite amount of information about Battery2. To represent the fact that no further information is available, the recursive model structure terminates on the second level with two *No-info* models.

In general, the *No-info* models have to differentiate further among knowledge limitations of the different agents involved. For example, the situation in which Battery1 does not have any information about how he is modeled by Battery2 is different from the situation in which Battery1 knows that Battery2 has no information about Battery1. The right-most *No-info^1* model in Figure 4 represents the fact that Battery1 has no information as to what Battery2 will do if he is incapacitated. The other two *No-info^1* models on the bottom of Figure 4 represent the fact that Battery1 has no information as to how Battery2 models Battery1, if Battery2 is not incapacitated.

Since the modeling structures, such as one in Figure 4, express the optimal choices in the agents' decision making situations recursively depending on the choices of the other agents, one can use dynamic programming to solve them in a bottom-up fashion. We elaborate on the solution procedure in more detail in [5], in which we have also outlined an analytical method of dealing with the *No-info* models. For the purpose of this study, however, we have designed and implemented a numerical method of solving the *No-info* models using logical sampling [14], according to the algorithm below.

Let $A_i = \{a_i^1, a_i^2, \ldots, a_i^n\}$ be the set of actions available to agent i. Let $P_i = [p_i^1, p_i^2, \ldots, p_i^n]$ be the probability distribution over the available actions, i.e., the conjecture as to agent i's action. The conjecture, P_i, is calculated based on frequencies, $F_i = (f_i^1, f_i^2, \ldots, f_i^n)$, with which each action turns out to be optimal, as follows:

Procedure No-info-PDF
Input the payoff matrix M_i of agent i.
Output the probability distribution, P_i,
over the actions of agent i.
begin
$F_i \leftarrow (0, 0, \ldots, 0)$
$N \leftarrow 0$
for each probability distribution, P_N, in the set of sampled probability distributions comprising the *No-Info* model,
Multiply probability distribution P_N by M_i.
Select an action a_i^k which has the maximum expected utility.
$f_i^k \leftarrow f_i^k + 1$
$N \leftarrow N + 1$
end for
$P_i \leftarrow [f_i^1/N, f_i^2/N, \ldots, f_i^n/N]$
return P_i
end

The dynamic programming bottom-up solution starts at Level 2. Using the sampling algorithm, with the sampling density of 0.1, on the second level of the recursive model structure, reveals that, if Battery2 has both short and long range interceptors, the probability distribution over Battery2's actions becomes $[0.00, 0.00, 0.00, 0.08, 0.21, 0.71]$ for interception of missiles A through F, respectively. Thus, if Battery2 has all of the ammunition, it will most likely attempt to intercept one of the closer threatening missiles. Similarly, the sampling No-info-PDF procedure invoked for the case of Battery2 lacking short range interceptors results in a $[0.70, 0.22, 0.08, 0.00, 0.00, 0.00]$ distribution over Battery2's limited actions of intercepting the missiles A through F, respectively. Thus, if Battery2 has no short range ammunition, it would be irrational for it to attempt to intercept any of the closer threats.

The above distributions over the models of Battery2 are given the weight corresponding to the probability of the model they resulted from, and are combined with the third *No-info*[1] model: (0.40 × [0.00, 0.00, 0.00, 0.08, 0.21, 0.71] + 0.40 × [0.70, 0.22, 0.08, 0.00, 0.00, 0.00] + 0.20 × [1/6, 1/6, 1/6, 1/6, 1/6, 1/6]) = [0.31, 0.12, 0.07, 0.07, 0.11, 0.32]. The resulting distribution is Battery1's overall expectation of Battery2's actions, given the uncertainty as to Battery2's available ammunition.

Propagating these results into Level 1, the combined probability distribution describing Battery2's actions is used to compute the expected utilities of Battery1's alternative behaviors as follows:

U_A: $0.31 \times 28.0 + 0.12 \times 42.9 + 0.07 \times 41.3 + 0.07 \times 50.8 + 0.11 \times 52.4 + 0.32 \times 55.9 = 43.84$
U_B: $0.31 \times 42.6 + 0.12 \times 23.6 + 0.07 \times 38.5 + 0.07 \times 47.9 + 0.11 \times 49.5 + 0.32 \times 53.0 = 44.54$
U_C: $0.31 \times 39.6 + 0.12 \times 37.1 + 0.07 \times 19.4 + 0.07 \times 44.9 + 0.11 \times 46.5 + 0.32 \times 50.0 = 42.43$
U_D: $0.31 \times 44.8 + 0.12 \times 42.2 + 0.07 \times 40.6 + 0.07 \times 28.1 + 0.11 \times 51.6 + 0.32 \times 55.2 = 47.20$
U_E: $0.31 \times 43.1 + 0.12 \times 40.5 + 0.07 \times 38.9 + 0.07 \times 48.4 + 0.11 \times 29.4 + 0.32 \times 53.5 = 44.52$
U_F: $0.31 \times 46.0 + 0.12 \times 43.4 + 0.07 \times 41.8 + 0.07 \times 51.3 + 0.11 \times 52.8 + 0.32 \times 35.4 = 43.23$

Thus, if Battery1 is rational, he will attempt to maximize his own expected utility and prefer to intercept MissileD first. This action is optimal, given Battery1's belief that Battery2 has both short and long range interceptors with probability 40%, that Battery2 has only long range interceptors with probability 40%, and that Battery2 has malfunctioned with probability 20%. According to the planning and execution cycle, the interceptor will be launched toward the selected missile, and the decision-making process will be repeated until there are no missile threats, or until Battery1 runs out of interceptors.

4.2. Bayesian belief update

In an uncertain and ever-changing situation, such as the defense scenario described above, it would be useful for Battery1 to be able to recognize that Battery2 is out of short range interceptors, and to adjust accordingly. Specifically, Bayesian learning [11, 14] can be used to update agent's belief about the other agent by changing the probability associated with the other agent's models based on their observed behavior.

The belief updating procedure consists of four steps:

Step 1: Identify others' possible models in one's knowledge base.

Step 2: Assign prior probabilities to others' models for initial belief (assign uniform probabilities if no prior information is available).

Step 3: Observe the other agents' behavior at a given time point.

Step 4: Update the beliefs about others' by using Bayesian learning and store the posterior beliefs in one's knowledge base. Go to **Step 3**.

To explain the **Step 4** in detail, suppose that agent i has a number of alternative models, $M_{ij} = \{M_{ij}^1, \ldots, M_{ij}^n\}$, that he uses to predict agent j's ($i \neq j$) behavior. The agent i's belief about agent j's model M_{ij}^k is denoted by $BEL(M_{ij}^k)$, which will be represented as a probability assigned to this model. Let $A_i = \{a_i^1, a_i^2, \ldots, a_i^n\}$ be the set of actions available to agent i. The prior probability that model M_{ij}^k is correct is $P(M_{ij}^k)$. Now, suppose that agent i observed that agent j executed an action a_j^l. Agent i can now update the probability assigned to the model M_{ij}^k as:

$$BEL(M_{ij}^k) \cong P(M_{ij}^k|a_j^l) = P(a_j^l|M_{ij}^k)P(M_{ij}^k)/P(a_j^l) \quad (3)$$

where
- $P(M_{ij}^k|a_j^l)$: the probability of model M_{ij}^k, given the observed action a_j^l;
- $P(a_j^l|M_{ij}^k)$: the probability that agent j would choose the action a_j^l if M_{ij}^k is an accurate model of agent j;
- $P(M_{ij}^k)$: the prior probability assigned to the given model;
- $P(a_j^l)$: the probability of this particular observed data.

In this formula, $P(a_j^l)$ is a normalization constant, resulting from the condition that probabilities of all models sum up to 1. $P(a_j^l|M_{ij}^k)$ is already known from the solution of the existing recursive model structure. The above belief update can be performed incrementally during a sequence of interactions, based on the other agents' observed behaviors.

In our scenario, Battery1 has three alternative models of Battery2 as described above. Consider the case of Battery1 observing that Battery2 attempted to intercept MissileF, as depicted in Figure 5 (a), in which interceptor 2 has been launched by Battery2. The beliefs about models M_{12}^1, M_{12}^2, and M_{12}^3 are initially assumed to be 0.4, 0.4, and 0.2, respectively (see Figure 4). The belief update about Battery2's model proceeds as follows:

$BEL(M_{12}^1) \cong P(M_{12}^1|\mathrm{F}) = P(\mathrm{F}|M_{12}^1)P(M_{12}^1)/P(\mathrm{F}) = (0.71 \times 0.40)/0.316 = 0.894$
$BEL(M_{12}^2) \cong P(M_{12}^2|\mathrm{F}) = P(\mathrm{F}|M_{12}^2)P(M_{12}^2)/P(\mathrm{F}) = (0.00 \times 0.40)/0.316 = 0.000$
$BEL(M_{12}^3) \cong P(M_{12}^3|\mathrm{F}) = P(\mathrm{F}|M_{12}^3)P(M_{12}^3)/P(\mathrm{F}) = (0.16 \times 0.20)/0.316 = 0.106$

where
- M_{12}^1: the model of Battery2 equipped with short range and long range interceptors;
- M_{12}^2: the model of Battery2 with only long range interceptors;
- M_{12}^3: the model according to which Battery2 has been damaged or incapacitated;

and with the probability distributions representing Battery2's expected actions of [0.0, 0.0, 0.0, 0.08, 0.21, 0.71], [0.70, 0.22, 0.08, 0.0, 0.0, 0.0], and [0.16, 0.16, 0.16, 0.16, 0.16, 0.16] computed based on the models M_{12}^1 through M_{12}^3, respectively.

Battery1's belief about Battery2's having both short and long range ammunition has thus increased to 89.4%, and the beliefs about other two alternatives have decreased. In other words, from the fact that Battery2 chose to intercept missile F, Battery1 was able to infer that, with high probability, Battery2 has both long and short range missiles.

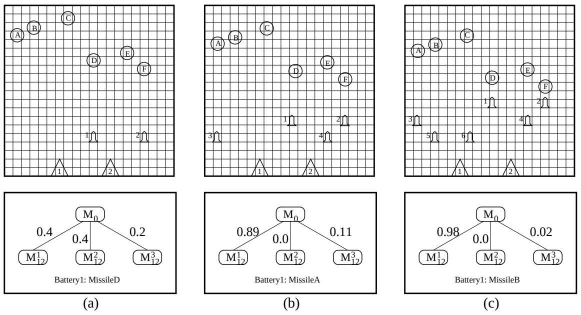


Figure 5: Battery1's belief update about Battery2. In this scenario, Battery2 has both short and long range missiles.

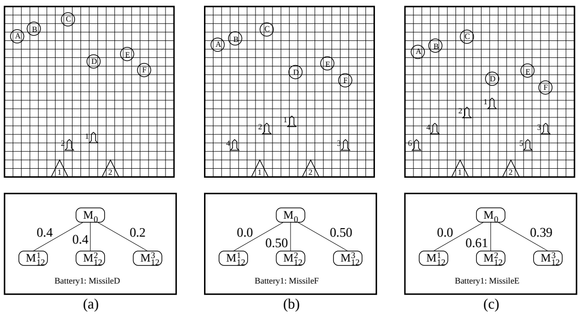


Figure 6: Battery1's belief revision about Battery2. In this scenario, Battery2 has only long range missiles.

Figures 5 and 6 illustrate the process of Battery1's updating his beliefs in two cases. In Figure 5, Battery2 is equipped with both short and long range interceptors, and has launched interceptors 2, 4, and 6. In

Figure 6, Battery2 has only long range interceptors, also numbered 2, 4, and 6. M_0 is Battery1's own payoff matrix in each figure. M^1_{12}, M^2_{12}, and M^3_{12} are the alternative models of Battery2, and Battery1's initial belief about Battery2 is shown in (a). Battery1's beliefs after the first interceptor launch, and after the second launch, are shown in (b) and (c), respectively. If Battery2 has both kinds of missiles, Battery1's belief about M^1_{12} converged to almost 100% after two interceptors launched. But, if Battery2 was equipped with only long range missiles, Battery1's belief about M^2_{12} changed less radically. In this case Battery1 remained unsure whether Battery2 had only long range missiles or whether he malfunctioned. The Bayesian belief update can enhance the quality of coordination during subsequent interactions, by correctly recognizing the models of the other agents from their behavior.

5. Experimental results

The anti-air defense simulator[3] is written in Common LISP and built on top of the MICE simulator [4], running on a LINUX platform. Each defense unit was assigned to one processor in a parallel and distributed environment. In the experiment we ran, each of two defense units could launch three interceptors, and were faced with an attack by six incoming missiles. We put all of the trials 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 size of each missile was constant during the experiment and the warhead sizes were comparable. In this experiment, the warhead sizes were $470, 410, 350, 370, 420, 450$ for missiles A through F, respectively. Third, for RMM team and RMM with learning team, both batteries were assumed to have short and long range interceptors with the belief 0.8 or to be incapacitated with the belief 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, during Bayesian belief update, each agent could identify the target aimed at by the other agent in the previous time step.

Our experiment was aimed at determining the quality of modeling and coordination 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 selected 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 $(1 - P(H)) \times W$ to the total damage. If a missile was not targeted, it contributed all of its warhead size value to the 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:[4] selection by human.
- RMM: selection by RMM without belief update.
- RMM with learning: selection by RMM with belief update.

The random agents were included to provide the worst base-line case of the system performance in our experiment. The performance achieved by independent agents was included to show what coordination can be expected when agents maximize but do not model each other. It turned out that the ways that human agents choose a missile were different and sometimes quite arbitrary. For example, some of our human subjects shot only the three missiles coming to their own side by dividing the grid world into left and right sides. They sometimes had difficulties in splitting the screen when the missiles were clustered at the center area, which led to much duplicated effort. Others tended to choose missiles with the largest missile size. Still others tried to consider the multiplication of the missile size and the hit probability, but did not model the other agent appropriately.

5.1. Performance assessments

We experimented with the above policies to understand the agent interactions in two groups: the agent teams with the same policy and the mixed agent teams with different policies.

As shown in Figures 7 and 8, we found that the all-RMM team outperformed the human and independent teams. The average number of selected targets by RMM after 100 trials is 5.49 ($\sigma_{\bar{X}}$[5] $= 0.05$), compared to 4.89 ($\sigma_{\bar{X}} = 0.08$) for the independent team

[3]For the demonstration of the anti-air defense domain, refer to the Web page at http://dali.uta.edu/Air.html.

[4]We should remark that our human subjects were simply 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 substantially better than our subjects. However, the defense doctrine remains classified, and was not available to us at this point.

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

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 is much less than that of the independent team ($732.0, \sigma_{\bar{X}} = 37.1$) and that of the all-human team ($772.0, \sigma_{\bar{X}} = 36.3$). The all-RMM with learning team showed better performance than the all-RMM team. The average number of selected targets was 5.82 ($\sigma_{\bar{X}} = 0.04$) and the total expected damage was 332.6 ($\sigma_{\bar{X}} = 18.1$).

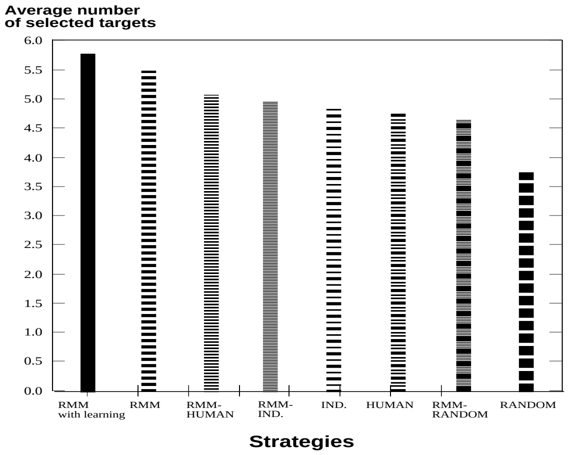


Figure 7: Average number of selected targets (over 100 runs).

The human performance was very similar to the performance of independent agents. The most obvious reason for this was that humans tended to depend on their intuitive strategies for coordination, and, in this case, found it hard to engage in deeper, normative, decision-theoretic reasoning. The common reason for the lower score of human teams was the choice of the greatest threat simultaneously–they made redundant efforts to attack the same missile. This, again, suggests that human agents attempted to minimize their expected damage but did not model the decision making of the other human very well, while RMM agents were rationally optimizing given what they expected of the other agent. The performance of the RMM and RMM with learning team was not perfect, however, since the agents were equipped with limited and uncertain knowledge of each other.

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

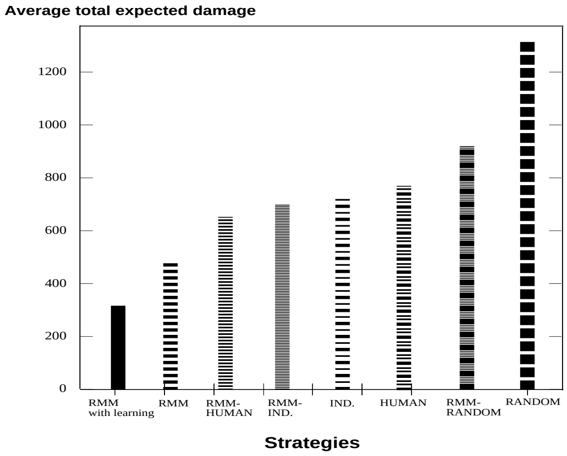


Figure 8: Average total expected damage (over 100 runs).

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.[6] In the experiment in which the number of selected targets was measured (Figure 7), $F(5, \infty) = 10.31$, $p < 0.01$. 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 by the RMM with learning team vs. the other non-human teams, t tests were performed. The results show that the RMM with learning team was better than any other team with the probability of 99% (0.01 level of significance).

The above results show that the recursive modeling allows RMM agents to improve the performance of target selection when paired with any other agent. A particularly promising facet of these results is that the recursive model structure (Figure 4) seems to be a robust mechanism for modeling and coordination with the human subjects, and not only among RMM agents.

[6] In our preliminary experiment, there are 4 pairs of all-human team and RMM-Human team. The results for humans are only suggestive and couldn't be evaluated statistically due to the small number of human participants. We will conduct more exhaustive experiments in future work.

6. Related work

The approaches related to the modeling and coordination in the multiagent environment are in the area of multiagent plan recognition and plan coordination. In work on plan recognition [13, 17], the objective is to enable an agent to model or recognize the other agents through observation to anticipate the other agents' future action, given prior knowledge about these agents. In these approaches, an agent usually compares a pre-calculated plan, or a protocol, with the on-going situation, and then chooses his next action accordingly. The process of mental-state recognition [13] assumes the correct and complete knowledge of the plans of the other agents, and it does not represent uncertainty that might be present in real-world domains. As Tambe and Rosenbloom [17] pointed out, ambiguities may persist when an agent must infer unobserved actions and intentions.

Multiagent coordination without communication has been dealt with in [15] and [8]. Sen and Sekaran [15] used a particular reinforcement learning methodology and concentrated on the learning classifier systems where agents share no problem-solving knowledge. However, the reinforcement learning method needs a super-agent for the external reward and requires numerous training cases to converge on the desirable entries. Mor et al. [8] handled the opponent's modeling in the complexity of polynomial time, based on game-theoretic techniques. They used a deterministic finite automaton to represent an agent's strategy during a learning process leading to equilibrium.

As far as belief change, Isozaki and Katsuno [7] estimated others' beliefs from observation by using a non-standard structure for modal logic. Their assumptions of belief estimation, i.e., disregard of messages, simplification of observations, and so on, are similar to the assumptions of our approach. However, they did not show the concrete example to validate their algorithm and did not provide the overall procedure of belief changes in the changing world. Brafman and Tennenholtz [1] define the agent as a qualitative decision maker with beliefs, preferences, and decision strategy. By using three components, the agent can predict other agent's future behavior and decide his own behavior based on the ascribed belief. In their method, the agent's ascribed mental state changes according to new observations. Comparing with our approach, they use the discrete descriptions of mental state, but our belief updating procedure uses a probability distribution based on the Bayesian theory.

7. Conclusion and further research

This paper presented a study in modeling and coordination in the multiagent distributed environment of anti-air defense. This investigation implies that the Recursive Modeling Method can be applied to high-level tactical decision making for minimizing overall damage. It turned out that it was relatively simple to quantify the missile threats and the hit probability to generate fairly realistic payoff matrices. Based on threat evaluation, we implemented the RMM agents to model the other agents and to follow the principle of maximum expected utility. Through Bayesian learning, the agents could increase the quality of the coordinated decision-making achieved within RMM framework. Thus, a rational agent could recognize the models of the other agents correctly and coordinate with them effectively in unexpectedly changing circumstances. We anticipate that the approach presented will also be applicable to other decision problems that require modeling, coordination, and belief change.

The most promising conclusion of our study is the high quality of the coordinated decision making achieved by the RMM agents. This, in turn, suggests that modeling other agents, including humans, as rational decision-makers is a promising and viable approach to achieving plan recognition and coordination. Also, the relatively high quality achieved by the mixed human/RMM team shows how a nested decision-theoretic modeling can provide for human-machine coordination without pre-established protocols that the human would have to learn and follow during the interaction.

In other experiments, we also implemented RMM in the predator-prey pursuit environment. There, the payoff matrices the agents used were 5-dimensional. The quality of coordination among the RMM agents for this domain is slightly inferior to that of human agents (http://dali.uta.edu/Pursuit.html). We also applied Bayesian belief update to the pursuit domain. In our future research we will investigate the influence of messages on the quality of coordination and belief update.

References

[1] R. I. Brafman and M. Tennenholtz. Modeling agents as qualitative decision makers. To appear in *Artificial Intelligence*, 1997.

[2] E. Charniak, C. K. Riesbeck, D. V. McDermott, and J. R. Meehan. *Artificial Intelligence Programming*. Lawrence Erlbaum Associates, Publishers, Hillsdale, New Jersey, 1987.

[3] J. Doyle and T. Dean. Strategic directions in ar-

tificial intelligence. *AI Magazine*, 18(1):87–101, 1997.

[4] E. H. Durfee and T. A. Montgomery. MICE: A flexible testbed for intelligent coordination experiments. In *Proceedings of the 1989 Distributed AI Workshop*, pages 25–40, Sept. 1989.

[5] P. J. Gmytrasiewicz and E. H. Durfee. A rigorous, operational formalization of recursive modeling. In *Proceedings of the First International Conference on Multi-Agent Systems*, pages 125–132, Menlo Park, 1995. AAAI Press/The MIT Press.

[6] R. A. Howard and J. E. Matheson. Influence diagrams. In R. A. Howard and J. E. Matheson, editors, *Readings on Principles and Applications of Decision Analysis*, pages 721–762. Strategic Decisions Group, Menlo Park, CA, 1984.

[7] H. Isozaki and H. Katsuno. A semantic characterization of an algorithm for estimating others' beliefs from observation. In *Proceedings of the National Conference on Artificial Intelligence*, pages 543–549, Portland, Oregon, Aug. 1996.

[8] Y. Mor, C. V. Goldman, and J. S. Rosenschein. Learn your opponent's strategy (in polynomial time)! In G. Weiß and S. Sen, editors, *Adaptation and Learning in Multi-Agent Systems — IJCAI'95 Workshop, Lecture Notes in Artificial Intelligence*, pages 164–176. Springer, New York, 1996.

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

[10] S. Noh and P. J. Gmytrasiewicz. Bayesian belief update in antiair defense. In *The Workshop of Machine Learning for User Modeling of the Sixth International Conference on User Modeling*, June 1997.

[11] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufman, San Mateo, CA, 1988.

[12] K. L. Poh and E. J. Horvitz. A graph-theoretic analysis of information value. In *Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence*, San Mateo, CA, 1996. Morgan Kaufmann.

[13] A. S. Rao and G. Murray. Multi-agent mental-state recognition and its application to air-combat modelling. In *Proceedings of the 13th International Distributed Artificial Intelligence Workshop*, pages 283–304, 1994.

[14] S. J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice-Hall, Englewood Cliffs, New Jersey, 1995.

[15] S. Sen and M. Sekaran. Multiagent coordination with learning classifier systems. In G. Weißand S. Sen, editors, *Adaptation and Learning in Multi-Agent Systems — IJCAI'95 Workshop, Lecture Notes in Artificial Intelligence*, pages 218–233. Springer, New York, 1996.

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

[17] M. Tambe and P. S. Rosenbloom. Architectures for agents that track other agents in multi-agent worlds. In M. Wooldridge, J. P. Müller, and M. Tambe, editors, *Intelligent Agents II: Agent Theories, Architectures, and Languages*, pages 156–170, Berlin, 1996. Springer.

[18] M. P. Wellman and J. Doyle. Preferential semantics for goals. In *Proceedings of the Ninth National Conference on Artificial Intelligence*, pages 698–703, July 1991.