Intelligent Command and Control Agent in Electronic Warfare Settings

Sanguk Noh,[1,*] Unseob Jeong[2,†]
[1]School of Computer Science and Information Engineering, The Catholic University of Korea, Bucheon 420-743, Republic of Korea
[2]Agency for Defense Development, Yuseong, P.O.Box 35-4, Daejeon 305-600, Republic of Korea

This paper investigates the autonomous decision-making process of threat detection, classification, and the selection of alternative countermeasures against threats in electronic warfare settings. We introduce a threat model that represents a specific threat pattern and also present a methodology that compiles the threat into a set of rules using soft computing methods. This methodology, which is based upon the inductive threat model, could be used to classify real-time threats. Furthermore, we calculate the expected utilities of countermeasures that are applicable given a situation and provide an intelligent command and control agent with the best countermeasure to threats. We summarize empirical results that demonstrate the agent's capabilities of detecting and classifying threats and selecting countermeasures to them in simulated electronic warfare settings.

1. INTRODUCTION

To counter-threats in electronic warfare environments, a command and control agent needs to detect, classify, and autonomously execute countermeasures against such threats for ensuring continual functionality despite potential danger. This paper investigates the whole decision-making process of threat detection, classification, and the selection of alternative countermeasures against threats. For a threat detection and classification, our agents advocate soft computing techniques,[1] and for the decision of countermeasures to the threats, they adopt a decision-theoretic architecture. Our agents thus make use of both architectures of learning and decision theory.[2]

Autonomous situation awareness perceives information about dynamically changing environment and accumulates the information to knowledge bases. Its ultimate step is to analyze the knowledge to predict what will happen in imminent future states. The process comes to involve tracking and identifying the state of a complex-distributed environment. It is not a reflexive response to an immediate

*Author to whom all correspondence should be addressed: e-mail: sunoh@catholic.ac.kr.
†e-mail: jeus@add.re.kr.

INTERNATIONAL JOURNAL OF INTELLIGENT SYSTEMS, VOL. 25, 514–528 (2010)
© 2010 Wiley Periodicals, Inc. Published online in Wiley InterScience
(www.interscience.wiley.com). • DOI 10.1002/int.20413

environment but a complex intelligence to make the knowledge operational. Autonomous situation awareness, thus, is a more critical component of adaptive knowledge formulation in urgent situations. It is widely applicable in areas such as battleground scenarios, traffic situations, and any kinds of disaster situations, e.g., fires, earthquakes, radiation accidents, and so on.[3–6]

We propose a threat detection and classification mechanism through soft computing algorithms, i.e., inductive decision-tree algorithms,[7] naïve Bayesian classifier,[8,9] and neural networks.[1] To identify threats that our agents face, we endow them with a tapestry of reactive rules.[4] The reactive rules are constructed by compiling threat systems and their attributes into state-action rules. The compilation process exploits the regularities of threats, if any, and enables our agents to detect them. The compiled rules performed offline can be obtained from soft computing algorithms, which use the threat information as their inputs. Furthermore, it is desirable that each of the compilation methods should be assigned a measure of performance that compares it to the benchmark. The various compilations available constitute a spectrum of approaches to make identifications and detections under various attacks in electronic warfare settings, and these compilations enable our agents to be aware of situations.

To distinguish dangerous situations from safe situations, we extract features from various types of threats in the simulated air combat scenarios using soft computing techniques. Applying soft computing algorithms for rule extraction has been used to detect specific phenomena in many domains[10,11] but, to our best knowledge, it might be the first attempt for the detection of threats at military scenarios. In our framework, the soft computing algorithms compile the instances of the threat system and their attributes into a set of reactive rules. Our approach allows us to model threat systems in battlefield situations and avoids critical situations that might be irrevocable. We present an inductive model of threat systems, and our compilation methods can appreciably shorten the response time between a condition occurrence and its recognition.

The next step, further, is to equip our command and control agent with the ability to dynamically and rationally select countermeasures against threats. Our agent will follow the decision theory,[2] which calculates the expected utilities of alternatives. The agents will finally succeed in completing their tasks by executing the best countermeasure, which has the maximum expected utility. Since the properties of electronic warfare environments are unforeseen, partially accessible, and continuously changing, the protocol-based approaches could not be applied to our setting. Applying the decision theory to selecting the countermeasures at military scenarios might be the first attempt to our best knowledge, and it might be a robust approach in battlefield situations.

In the following section, we will show clearly factors that indicate the symptoms of various threat systems and then design the intelligent command and control agent, which is operational in electronic warfare settings. Section 3 describes our agent's decision-making process of threat detection, classification, and the optimal selection of alternative countermeasures. Section 4 validates our framework empirically and presents the experimental results. In conclusion, we summarize our result and discuss further research issues.

2. ANALYZING THREATS IN ELECTRONIC WARFARE SETTINGS

To improve the survivability of our agents in battlefield settings and to enable them to successfully perform their mission, we extract features from various threat systems and design a command and control agent that operates autonomously.[12,13]

2.1. Analyzing Threats

Our agents detect their potential threats through their sensors: radar, laser, and infra-red. The threats could be classified into "terminal" and "nonterminal" threats.[12,14,15] The terminal threats are intended to directly shutdown our agents, whereas the nonterminal threats are preliminary operations to enhance the capability of the terminal threat systems. The terminal threats consist of "static" and "mobile" lethal objects. The static terminal threats are antennas, power lines, buildings, and so on. On the other hand, the mobile terminal threats are missiles, guns, and rockets. The nonterminal threats include searching, tracking, and taking electronic countermeasures against communication systems.[16–18]

Some of the attributes contained in agents' knowledge bases (KBs) when they interact in battlefield scenarios are summarized in Table I. The attributes in Table I are selected to effectively distinguish between the threat types and the threat levels among all possible attributes. The attributes that should be considered regarding the threats detected by radar sensors are predominant and can be easily picked up, whereas those of the threats identified through laser sensors are usually limited. Since infra-red (IR) sensors have no range information and strongly depend on atmospheric conditions,[15,19] the usage of IR sensors is restricted. However, the radar sensors are durable under all weather conditions and the attributes involving radar sensors are actively considered for the nonterminal and the terminal threats. As shown in Table I, the attributes acquired from radar sensors are radar frequency, pulse width, pulse power, and pulse repetition frequency. The other attributes, i.e., pulse repetition frequency and guidance type, characterize the terminal threats confirmed by laser sensors.

Table I. Relevant attributes in electronic warfare settings.

Receiver Types	Attributes	Values	Threat Types
Radar	Radar frequency	30–8,000 MHz	Nonterminal
	Pulse width	0.8–5 ms	
	Pulse power	10–500 KW	
	Pulse repetition frequency	1–666 KHz	
Radar	Radar frequency	8,000–40,000 MHz	Terminal
	Pulse width	0.1–0.8 ms	
	Pulse power	1–50 KW	
	Pulse repetition frequency	333–1,000 KHz	
Laser	Pulse repetition frequency	0.1–20 KHz	Terminal
	Guidance type	Range finder, Target designator, Beam rider	
Infra-red	Target coordination	x, y, z	Nonterminal/terminal

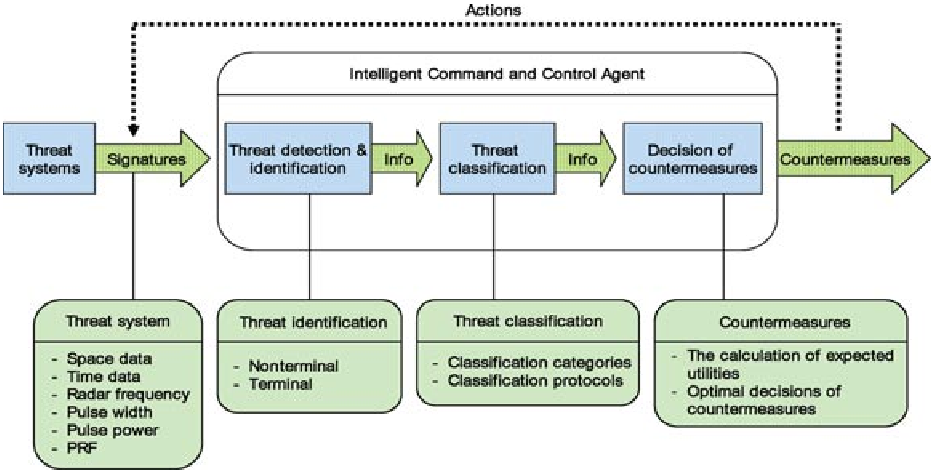

Figure 1. Decision-making process of threat identification, threat classification, and the selection of countermeasures against threats.

2.2. Designing Command and Control Agent

Our aim is to design autonomous agents that quickly respond to threat systems represented by the above attributes in Table I, while operating in real-time electronic warfare settings. The first step toward this end is to integrate the symptoms of threat systems and to detect and identify the threat systems themselves. We then classify the threats into terminal and nonterminal ones based upon categories compiled during offline. The final step of the command and command module is to dynamically decide the best countermeasure against threats using the computation of expected utilities in conjunction with online reasoning. The intelligent command and control agent that achieves our goal is illustrated in Figure 1.

For the intelligent command and control agent, we propose a brokering agent architecture that consists of (1) an information collecting and processing agent that gathers the signatures of threat systems, (2) an adaptive reasoning agent that detects threat systems upon the pre-compiled protocols, and (3) a decision-theoretic agent that finally executes the best countermeasure among alternatives. This architecture, as shown in Figure 1, allows our autonomous agents to quickly recognize a current situation using the precompiled protocols and to actively remove potential adversities with robust autonomy through the calculation of expected utilities. The fast report of the current situation and the rational decision of the countermeasures prepare the command and control agents for an urgent situation and provide them with autonomous capability without relying on the operation of human beings.

3. IDENTIFYING THREATS AND DECIDING COUNTERMEASURES

To inspire adaptability into our agents, we use soft computing algorithms,[1] i.e., naïve Bayesian classifier, inductive decision tree algorithms and neural networks, and compile the models of threat systems into a set of rules using them. To provide our agents with rationality, further, we use the decision theory[2] that combines preferences with probabilities, in case of selecting countermeasures.

3.1. Bayes Theorem and Induction Algorithms

Bayes rule examines whether or not a property observed as evidence belongs to a specific hypothesis (or class), given a set of data distribution. Bayes theorem[8,9] can be defined as follows:

$$P(h_j \mid x_i) = \frac{P(x_i \mid h_j)P(h_j)}{\sum_{j=1}^{m} P(x_i \mid h_j)P(h_j)} \tag{1}$$

where

- a set of observable attributes, $X = \{x_1, x_2, \ldots, x_n\}$;
- a set of hypotheses in a domain, $H = \{h_1, h_2, \ldots, h_m\}$;
- $P(h_j \mid x_i)$ is the posterior probability of the hypothesis $h_j, h_j \in H$, given that $x_i, x_i \in X$, is an observable event.

In our framework, the set of observable attributes, X, consists of the attributes, as described in Table I, and the hypotheses are a composite of which threat system is given and how the threat is effective to our agents. Given the set of data as evidence, Bayes rule allows us to assign probabilities of hypotheses, $P(h_j \mid x_i)$. Our agents compute $P(h_j \mid x_i)$ during online and set an alarm when the probability of a specific threat to them given input is greater than those of any other threat systems.

The decision tree approach such as ID3, C4.5,[7] and CN2[8] is to divide the domain space into classified regions, which are given by a set of classes $C = \{c_1, c2, \ldots, c_m\}$. The basic idea of the decision tree-based induction algorithms is to find out a set of ordered attributes, $X = \{x_1, x_2, \ldots, x_n\}$, which split the data sets into a correct classification with the highest information gains first. A decision tree has internal nodes labeled with attributes $x_i \in X$, arcs associated with the parent attributes and leaf nodes corresponding to classes $c_j \in C$.

The decision tree-based induction algorithms, thus, generate a tree representing a model of various threats to our agents in the simulated electronic warfare setting. Once the tree is built using training data, the optimal rules from the tree can be obtained and are applied to a new threat environment to determine whether any potential attack to our agents is made.

In supervised learning, artificial neural networks[1] find a function that approximates the training examples and infer the mapping implied by the examples. The output of the neural network, thus, shows the computational model of threat systems that our agents face and provides us with the understanding of the mapping from the attributes of the threats to the threat systems themselves.

3.2. Compilation of Threats into Rules

Our approach to make adaptive reasoning more reactive is similar in spirit to methods advocated in Ref. 20. The way of utilizing reactive rules is to rely on results accumulated during offline and to use these resulting rules during online. The offline computation can be used to summarize regularities given in a sample of situations. We choose to represent the found regularities as reactive rules dictating which consequences should be candidates for execution, depending on the circumstances.

Let S be the set of battlefield states that the adaptive reasoning agent can discriminate among, and let L be the set of compilation methods (machine learning algorithms) that the agent employs. Given a machine learning algorithm $l \in L$, a set of compiled decision-making rules of an adaptive reasoning agent is defined as

$$\rho_1 \colon S \to \{\text{threat system}\} \tag{2}$$

representing whether a specific threat occurs in the state $s \in S$. Thus, various machine learning algorithms compile the models of threat into different functions ρ_l, each enabling the agent to take advantage of regularities in the environment in a different way.

Given the compiled knowledge, further, we represent what amount of threats to our agents are made as follows:

$$\varpi \colon \mathrm{KB} \times S \to \{\text{threat level}\}. \tag{3}$$

In (3), KB is the set of compiled knowledge of the adaptive reasoning agent. The threat level can be defined as a set of vectors with *attributes*, as described in Table I, predictive *arrival time* to our agent, and *threat characteristics*, which is specific depending on the threat types, i.e., terminal and nonterminal. We generate the training examples for the learning algorithms from terminal and nonterminal threat environments, respectively.

3.3. Deciding Countermeasures against Threats

To be rational in a decision-theoretic sense, the agents follow the principle of maximum expected utility (PMEU).[2] We will show how PMEU can be implemented in the decision-making process of the selection of countermeasures under uncertainty. Our agents that are equipped with PMEU will select the most appropriate countermeasure to effectively remove threats.

We will use the following notation:

- a set of agents: $N = \{n_1, n_2, \ldots\}$;
- a set of actions of agent n_i, $n_i \in N$: $An_i = \{a_i^1, a_i^2, \ldots\}$; and
- a set of possible world states: $S = \{s_1, s_2, \ldots\}$.

The expected utility of the best action, α, of agent n_i, arrived at using the body of information E, and executed at time t, is given by[a]

$$EU(\alpha \mid E, t) = \max_{a_i^j \in A_{ni}} \sum_k P\left(s_k \mid E, t, a_i^j\right) U(s_k) \tag{4}$$

[a] Our notation follows Ref. 2.

where $P(s_k \mid E,t,a_i^j)$ is the probability that a state s_k will be obtained after action a_i^j is executed at time t, given the body of information E; and $U(s_k)$ is the utility of the state s_k.

For the purpose of formalizing the decision-making problem of selecting countermeasures against threats, we should model the probabilities and the utilities in (4). In our model, for example, the probability[b] that a countermeasure would be successful is assumed to depend on the jamming signal power, the useful signal power reflected, the distance between the radar and aircraft, and so on, when jamming countermeasures are executed. The utility that denotes the desirability of a resulting state after a countermeasure is executed can be assigned by a single number considering the type of receivers. We will give the concrete example of the computation of expected utilities with four countermeasures in the following section.

4. SIMULATION TESTS AND RESULTS

The experiments in this section are designed to evaluate (1) the performances of threat detection and classification and (2) those of the decision of countermeasures against threats. First, we generate and validate the compiled rules by applying them to simulated electronic warfare settings. We use WEKA (Waikato Environment for Knowledge Analysis)[22] for machine learning techniques. We measure an adaptive reasoning agent's performance in terms of the correctness of the threat classification. In the second experiment, we measure the decision-theoretic agent's performance in terms of the sum of expected utilities of the best countermeasures selected given a situation.

4.1. Generation of Simulation Data

To test our agent's performance in various models of multispectral threat data, we generated the simulation data using three different distributions, i.e., discrete uniform, Gaussian, and exponential distributions. Given the attributes, as described in Table I, the information-gathering agent collected the threat data acquired from the radar, laser, and infra-red receivers, respectively. The threat data through three distributions were generated within the range of attribute values. To endow our agent with the inductive models of threat data, then, the threats as training data were compiled into a set of rules. The resulting models of threats could get closer to the uncertain patterns of real threats, since these three distributions widely covered the possible distributions of threat data. Figure 2 shows a part of threat simulator, which represents the simulation data for *radar frequency* (RF) based upon discrete uniform, Gaussian, and exponential distributions.

In the experiment, the distribution of the attribute values was similar to an actual data set. The attribute values of radar frequency, for example, are depicted in Figure 2. The x-axis represents the range of *radar frequency (RF),* which actually

[b] Refer to our concurrent work[21] for details.

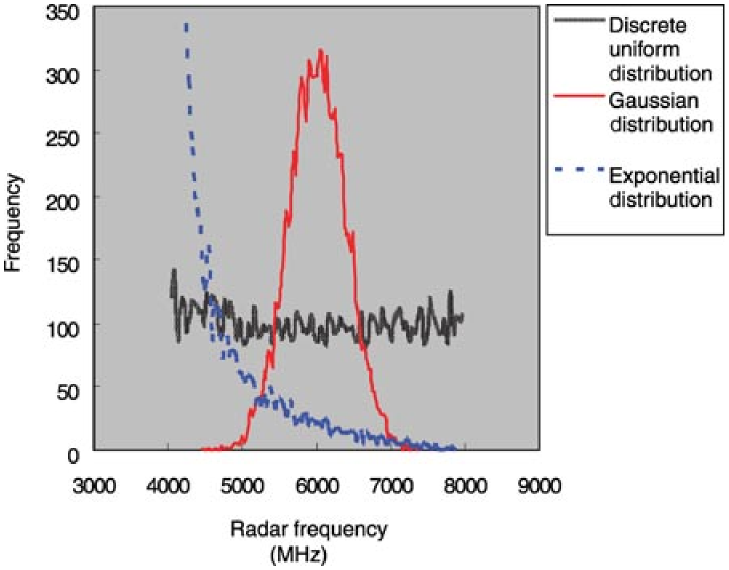

Figure 2. The three distributions of radar frequency (RF) attribute values for the class of middle- and long-range radar and track. The size of data set is 10,000.

ranges from 4000 to 8000 MHz, and the y-axis shows the number of training examples given distributions.

4.2. Learned Condition–Action Rules in Electronic Warfare Settings

To construct compiled rules for our agents, we used three machine learning algorithms: naïve Bayesian classifier, C4.5, and multilayer perceptron. Hence, the set of compilation methods L, as described in (2), was {naïve Bayes ($= l_1$), C4.5 ($= l_2$), multilayer perceptron ($= l_3$)}. For the naïve Bayesian classifier, the results are represented as rules specifying the probability of occurrence of each attribute value given a class,[8] in our case "threat system." C4.5 represents its output as a decision tree, and the trained result of multilayer perceptron is a mapping from the attributes of threats to the threat systems as a function. We now describe the agent's compiled knowledge by using the above learning algorithms.

For each target in an electronic warfare setting, the training data for radar and laser receivers were obtained as a tuple of attribute values described in Table I, and a class, for instance, "missile" as a mobile terminal threat and "long-range search" as a nonterminal threat. The numbers of classes for the terminal threats acquired from radar and laser receivers were three and four, respectively, and there were five classes and one class for the nonterminal threats received from radar and laser receivers, respectively. The instances consisting of the attribute values and the classes were fed into the learning algorithms as training data.

The learning algorithms l_1–l_3 then compiled the training data into a different set of rules $\rho_{l,}$ as defined in (2). The learned state-action rules take into account only the

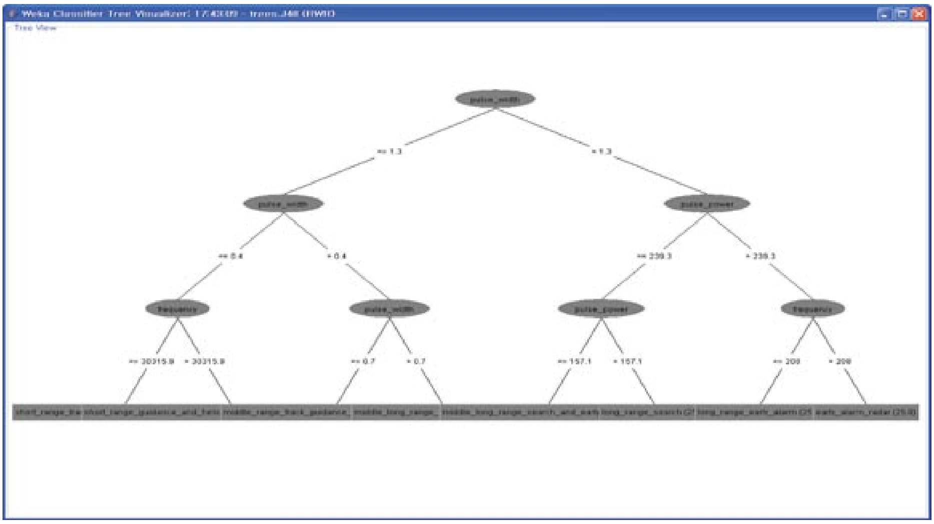

Figure 3. A learned decision tree obtained by C4.5 using WEKA.[20]

properties of one type of receivers at a time, not of the whole configuration. In other words, the threats acquired from the radar and the laser receivers were separately fed into the compilation methods. As an example, a decision tree obtained using C4.5 ($= l_2$) for these attributes is depicted in Figure 3. Based on the resulting decision tree for a nonterminal threat system, one of compiled rules in ρ_{l2} is "**if** pulse width >1.3 **and** pulse power >239.3 **and** radar frequency >208, **then** early_alarm_radar."

4.3. Performance of Compiled Rule Sets

To evaluate the quality of various rule sets generated by different learning algorithms, the performance obtained was expressed in terms of the correctness of the threat classification. As a first step, to find a meaningful size of the training set, which could guarantee the soundness of the learning hypothesis, we generated several sets of training examples using three different distributions, i.e., discrete uniform, Gaussian, and exponential distributions. As the number of the training examples increased, in general, the resulting performance improved sharply up to a certain point, after which performance leveled off. We found that the sufficient number of training instances was 100. For the radar receiver, the performance was the best in case of the normal distribution, and, for the laser receiver, the best performance was achieved, when the exponential distribution was used. The learning curves that represent the resulting performances (%) vs. the sizes of training examples for the radar receiver given normal distribution and the laser receiver given exponential distribution are depicted in Figures 4 and 5, respectively.

For radar receiver, the naïve Bayesian classifier learned the function of the classification of threat systems quickly, as shown in Figure 4. The best performance

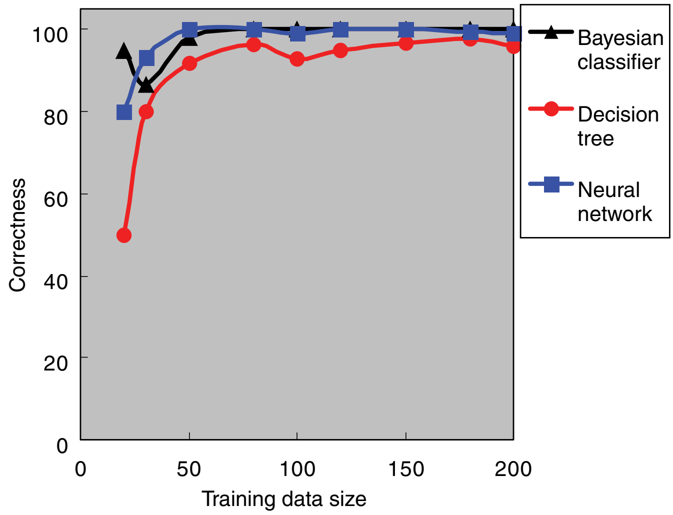

Figure 4. Resulting performances (%) vs. the training data size for radar sensors.

of naïve Bayesian classifier correctly classified 100%, whereas those of C4.5 and multilayer perceptron did 96% and 99%, respectively. On the other hand, for laser receiver, the best performance of 98.3% was achieved by the rules compiled using C4.5, as depicted in Figure 5. The performance obtained by decision-tree based induction algorithm C4.5 was better than those of naïve Bayesian classifier and

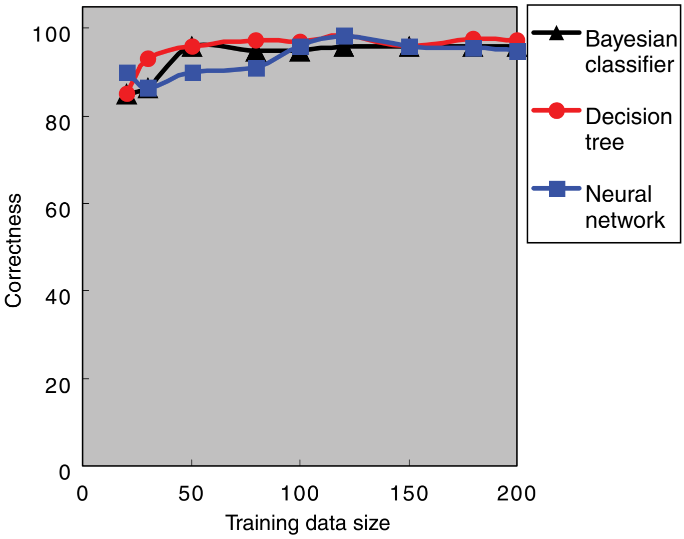

Figure 5. Resulting performances (%) vs. the training data size for laser sensors.

Table II. Performances of compilation methods for radar and laser receivers.

Receiver Types	Compilation Methods	Performances
Radar	Naïve Bayes	100.0 ± 0.00
	C4.5	95.0 ± 0.94
	Multilayer Perceptron	98.3 ± 1.34
ANOVA	72.41	
Laser	Naïve Bayes	95.8 ± 1.40
	C4.5	97.4 ± 1.35
	Multilayer Perceptron	95.8 ± 1.48
ANOVA	4.30	

multilayer perceptron. The best performance of naïve Bayesian classifier through the training data for laser receiver correctly classified 95.8%.

We applied the compiled state-action rules ρ_l, obtained by different learning methods l, into newly generated 10 sets of 100 scenarios and could get the performances of the learning methods, as described in Table II.

We analyzed the performance results in Table II using the standard analysis of variance (ANOVA) method. Since the computed values of $f = 72.41$ and $f = 4.30$ in ANOVA exceed 5.39 ($= f_{.01,2,27}$) and 3.32 ($= f_{.05,2,27}$) from the F distribution, we know that our agents, controlled by three different methods, in both of threat situations detected using radar and laser sensors were not all equally effective at the 0.01 level and 0.05 level of significance (i.e., the differences in their performance were not due to chance with probability of 0.99 and 0.95), respectively. In Table II, the average performance of our agent using naïve Bayesian classifier in a threat situation from radar receiver's perspective was slightly better than those of C4.5 and multilayer perceptron, whereas the agent with rules compiled by C4.5 outperformed the other agents with rules compiled by naïve Bayes and multilayer perceptron in a threat situation simulated through laser receiver.

4.4. Determining the Rational Choice of Countermeasures

As a simple example, let us consider an electronic warfare scenario. This scenario has a command and control agent confronting a specific threat. The mission of our agents is to autonomously decide and execute their countermeasures to a specific threat. The agent is assumed to be equipped with four countermeasures, $An_i = \{$*chaff, flare, RF jamming, IR jamming*$\}$. In this example scenario, our agent identifies a threat through only a missile-warning receiver (MWR), which is scenario 2 in Table III. According to the types of receivers, the countermeasures that can be applicable are limited, and, in this case, only the *flare* and *IR jamming* can be useful, as described in Table III.

Given the situation at hand, our agents following the decision theory should choose a countermeasure that maximizes their expected utility, as described in (4). First, the probabilities that each countermeasure would be successful can be acquired

Table III. Payoff matrix of utilities in electronic warfare settings.

	Type of Receivers			Utility Values of Countermeasures			
Scenarios	RWR	MWR	LWR	Chaff	Flare	RF Jamming	IR Jamming
1	○			0.393	–	0.551	–
2		○		–	0.393	–	0.551
3			○	–	–	–	–
4	○	○		0.393	–	0.551	–
5	○		○	0.393	–	0.551	–
6		○	○	–	–	–	–
7	○	○	○	0.393	–	0.551	–

RWR: Radar-warning receiver, MWR: Missile-warning receiver, and LWR: Laser-warning receiver.

through the computation in our concurrent work[21] as follows:

- $P(Result_{success}(flare) \mid Do(flare), E, t) = 0.781$;
- $P(Result_{success}(IRjam) \mid Do(IRjam), E, t) = 0.486$.

Second, the utility that denotes the desirability of a resulting state after a countermeasure is executed can be summarized in Table III. The utility values of RF jamming and IR jamming are greater than those of the chaff and flare, and their specific utility values can be obtained from the utility function, $1/(1 - e^{-\lambda x})$, where λ is the constant of 0.1 and x is a real value between 1 and 10, which is corresponding to one of the countermeasures. When no countermeasures are successful, the utility value that our agents can have is assumed to be 0.095, where the value of x is 1.

Thus, the expected utilities of the command and control agent's alternative countermeasures, as defined in (4), are

$$EU(flare \mid E, t) = 0.781 \times 0.393 + 0.219 \times 0.095 = 0.328,$$

$$EU(IRjam \mid E, t) = 0.486 \times 0.551 + 0.514 \times 0.095 = 0.317.$$

In this example scenario, thus, our command and control agents will take the action of the flare as their best countermeasure.

To evaluate the quality of the decision-making process of countermeasures against threats in electronic warfare settings, the resulting performance was expressed in terms of the cumulative expected utilities. The cumulative expected utilities are defined as the sum of expected utilities after 30 selections of countermeasures have been made. The average of the cumulative expected utilities through 10 sets of 30 selections was summarized in Figure 6.

In this experiment, the strategies for selecting the countermeasures are as follows:

- *α strategy:* the selection of the countermeasure that has the highest expected utility;
- *β strategy:* the selection of the countermeasure that has the highest probability representing its success, when it is executed; and
- *γ strategy:* the random selection of the countermeasure.

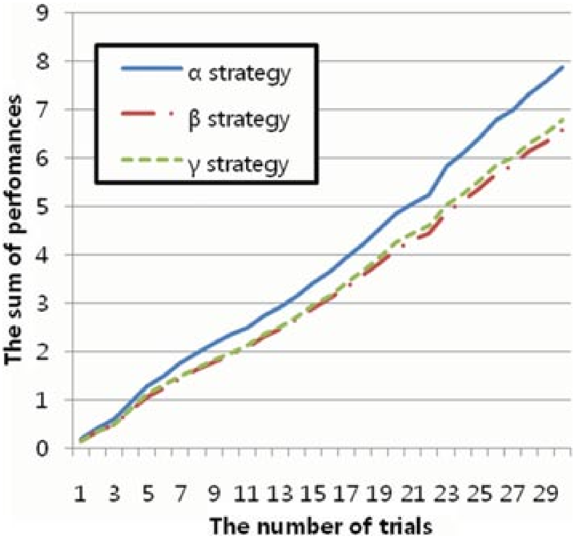

Figure 6. The sum of performances (expected utilities) vs. the number of trials for the selection of alternative countermeasures.

As we expected, in Figure 6, the performance achieved by our agents following the decision theory was better than that of the agent guided by the random selection strategy. The performance of β strategy was similar to that of the random selection strategy. Compared with the performance, 6.7959, of the random agents, the performance, 7.8707, of our agent was increased by 15.81%.

5. CONCLUSION

In time-critical settings, autonomous agents need to quickly recognize a given situation and to rationally react to it. Our work contributes to situation awareness, when robust autonomy is crucial. In this paper, we present a fully autonomous command and control agent in electronic warfare settings. From the command and control agent's perspectives, we showed the whole decision-making process of threat detection, classification, and the selection of alternative countermeasures against threats.

For the threat detection and classification, we analyzed threat systems into a set of attributes and formulated a compilation method that endows our agents with reactivity. The reactive rules compiled by machine learning algorithms exploited the regularities of domains. Our agents then were able to classify threats using the compiled rules. To be rational in dynamic electronic warfare settings, further, our agents were capable of choosing and executing countermeasures to threats, as maximizing their expected utilities.

We tested our agent's performance in simulated electronic warfare settings. The threat data in these settings were generated using discrete uniform, Gaussian, and exponential distributions, which got closer to real threats. The preliminary experiments revealed that the compiled rules were useful to accurately report the given situation and to mine specific patterns from complex situations, and the computation of the expected utilities made our agents rationally operational in dynamic environments.

As part of our ongoing work, we are performing a set of experiments with all possible configurations of threat systems and are implementing a threat simulator. We will integrate various threat systems into a unified battlefield scenario and continuously test our agent's rationality with a tapestry of scenarios. We hope to be able to reduce the total number of false alarms, to successfully remove threats through our future work, and to apply our framework to other time-critical domains.

Acknowledgment

This work has been supported by the Catholic University of Korea research fund granted in the program year of 2009 and by the Agency for Defense Development, Korea, under Grant UD070078ED "Autonomous Threat Detection and Classification," 2007. We would like to thank our students, Shinbong Kang and Hunwoo Park, for their help in implementing the threat simulator.

References

1. Zadeh LA. Fuzzy logic, neural networks and soft computing. Commun ACM 1994;37(3):77–84.
2. Russell SJ, Norvig P. Artificial intelligence: A modern approach, 2nd edition. Englewood Cliffs, NJ: Prentice-Hall; 2003. Chaps. 13–17.
3. Bryant DJ, Lichacz FMJ, Hollands JG, Baranski JV. Modeling situation awareness in an organizational context: military command and control. In: Banbury S, Tremblay S, editors. A cognitive approach to situation awareness: Theory and application. Burlington, VT: Ashgate Publishing; 2004. Chapt. 6.
4. Noh S. Autonomous situation awareness through threat data integration. In: Proc IEEE Int Conf on Information Reuse and Integration, 2007. pp 550–555.
5. Noh S, Gmytrasiewicz P. Flexible multi-agent decision-making under time pressure. IEEE Trans Syst Man Cybern Part A: Syst Humans 2005;35(5):697–707.
6. Patrick J, James N. A task-oriented perspective of situation awareness. In: Banbury S, Tremblay S, editors. A cognitive approach to situation awareness: Theory and application. Burlington, VT: Ashgate Publishing; 2004. Chap. 4.
7. Quinlan JR. C4.5: Programs for machine learning. San Francisco, CA: Morgan Kaufmann; 1993.
8. Clark P, Niblett T. The CN2 induction algorithm. Mach Learn J 1989;3(4):261–283.
9. Hanson R, Stutz J, Cheeseman P. Bayesian classification theory. Technical Report FIA-90-12-7-01; NASA Ames Research Center, AI Branch, 1991.
10. Kasabov N. Adaptation and interaction in dynamical systems: modeling and rule discovery through evolving connectionist systems. Appl Soft Comput 2006;(6):307–322.
11. Moshou D, Hostens I, Papaioannou G, Ramon H. Dynamic muscle fatigue detection using self-organizing maps. Appl Soft Comput 2005;5:391–398.
12. Advanced EW protection for maximum survivability. Northrop Grumman, AN/APR-39B(V)2 Suite of integrated sensors and countermeasures (SISCM).
13. Aircraft survivability equipment (ASE): ensuring lethality and dominance of Army aviation over tomorrow's battlefield. Association of the United States Army; July 2002.

14. Electronic systems—Defensive systems division, Northrop Grumman. Available at http://www.es.northropgrumman.com/. Accessed on March 5, 2010.
15. Heikell J. Electronic warfare self-protection of battlefield helicopters: A holistic view. Doctoral dissertation. Helsinki University of Technology, Helsinki, Finland; 2005.
16. Use of helicopters in land operations. STANAG 2999 HIS (NATO ATP-49(A)); 1992.
17. Kennedy LD, Patterson CR, Munshower DC. F/A-18 electronic warfare suite cost and operational effectiveness analysis methodology: phase 1– radio-frequency countermeasures. Johns Hopkins APL Technical Digest 1997;18(1);59–68. Also available at http://techdigest.jhuapl.edu/td1801/kennedy.pdf. Accessed on March 5, 2010.
18. Thompson LB. Army force protection: Helicopters can't wait. News article, April 21, 2006; Available at http://www.lexingtoninstitute.org/army-force-protection-helicopters-cant-wait. Accessed on March 5, 2010.
19. Zhongwen H, Zongxiao W. Sources and techniques of obtaining National Defense Science and Technology Intelligence. Beijing, People's Republic of China: Kexue Jishu Wenxuan; 1991. Available at http://www.fas.org/irp/world/china/docs/. Accessed on March 5, 2010.
20. Rao AS, Georgeff MP. BDI agents: from theory to practice. In: Proc First Int Conf of Multiagent Systems, 1995. pp 312–319.
21. Park SR, Kim S, Noh S, Kim K, Go E, Jeong U. A study on the effectiveness of active/passive/active-passive RF jamming in electronic warfare. In proceedings of KICS Fall Conference, IC-4. Nov 2008.
22. Witten IH, Frank E. Data mining: practical machine learning tools and techniques, 2nd edition. San Francisco, CA: Morgan Kaufmann; 2005.