Utility-Based Sequential Decision-Making In Evidential Cooperative Multi-Agent Systems

Galina Rogova
Encompass Consulting
Honeoye Falls, NY, U.S.A.
rogova@rochester.rr.com

Carlos Lollett
Computer Science and Engineering Department
University at Buffalo
Buffalo, NY, U.S.A.
clollett@acsu.buffalo.edu

Peter Scott
Computer Science and Engineering Department
University at Buffalo
Buffalo, NY, U.S.A.
peter@cse.buffalo.edu

Abstract - *This paper presents a new approach to building utility-based models of decision-making in time-constrained situations with limited resources. A particular hierarchical homogenous multi-agent architecture has been considered. The proposed system combines agents' beliefs within the framework of evidence theory and after each observation maps the current set of cumulative pignistic probabilities into one of two actions: "defer decision" or "decide hypothesis i". The system maximizes the expected utility of delayed decisions minus cost. The process of system adaptation to the environment is guided by reinforcement learning. The utilities-from-experts problem is simplified by learning utilities directly from feedback on the quality of the decisions. The results of a case study are presented.*

Keywords: Decision utility, Reinforcement learning, Distributed systems, Multi-agent systems, Sequential decision-making, Evidence theory.

1 Introduction

The goal of the research described in this paper is to investigate the problem of sequential decision making with limited resources in time-constrained situations in cooperative multi-agent systems. Timely action in time-constrained situations is needed in many real-world applications, for example, in medical decision making in the emergency room when a patient's condition is deteriorating but the optimal treatment decision can be made with complete confidence only after expensive, time-consuming tests. The same considerations obtain in target recognition when additional observations can improve the quality of recognition and help avoid errors but at the same time the cost of delay is very high and there are competing demands on the sensors required for the additional observations. In general, sequential decision making with constrained resources can be considered as a part of resource management in which a decision-maker performs a trade-off between benefits of additional observations and cost of delaying decision and its resulting action.

In this research based on our previous studies [1-3], we consider a hierarchical homogeneous multi-agent system in which agents with a common internal structure, including domain knowledge, a common set of hypotheses to be considered, and a common procedure for assigning a level of belief to each hypothesis, are able to extract different features from the environment but are unable to communicate directly with one another. They passively acquire information from observations at discrete times $t = 1,2,...t^*$, $t^* \le T$, where t^* is the time of selection of any particular hypothesis and T is the deadline by which a classification decision is required. At each time t each agent produces beliefs in each hypothesis under consideration and transmits these beliefs to the Fusion Center (FC). The FC combines of all the beliefs obtained from the agents up to and including t within the framework of evidence theory [4,5] and produces cumulative pignistic probabilities for each hypothesis. The decision-maker then maps the current set of cumulative pignistic probabilities into one of two actions: "defer decision" or "decide hypothesis i".

The process of system adaptation to the environment is guided by reinforcement learning. Reinforcement learning is the strategy by which an agent learns

behavior through recalling the reinforcements received during trial-and-error interactions with a dynamic environment [6]. In contrast with supervised learning, where the environment serves as a teacher that explicitly provides the desired decision as a feedback, reinforcement learning does not rely on "exemplary supervision" or complete models of the environment. In learning through trial-and-error, the system makes a decision and the environment returns a reinforcement signal reflecting the degree of correspondence of the selected hypothesis to the long-term goal. Our goal is to teach the system to minimize the average loss associated with a wrong decision by taking advantage of agents' collective knowledge and the feedback from the environment. Loss is defined in terms of utility and cost, as will be detailed in the next section. During the learning process the decision-maker, upon deciding hypothesis i at time t^*, presents the decision to the environment, which returns a real reinforcement signal reflecting both the utility of the decision and the decision latency t*. The reinforcement signal is then fed back and distributed among the lower level agents modeled as reinforcement learning neural networks [3], which utilize it to incrementally improve their policy of mapping observed features into beliefs.

The proposed system architecture is presented in Figure 1

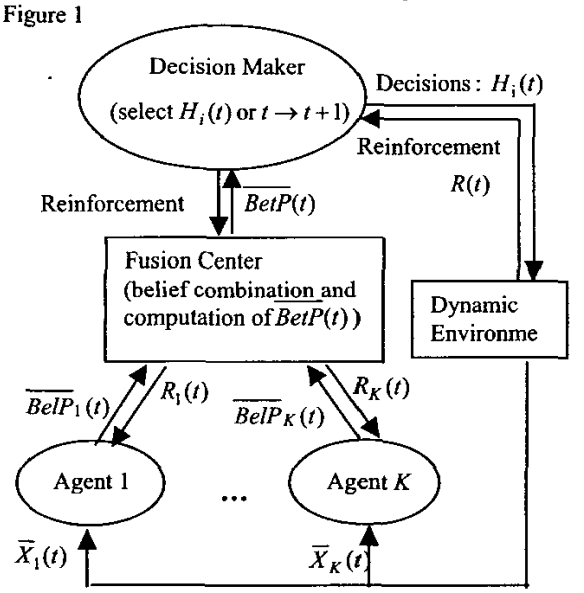

Figure 1. System architecture

This paper is focused on modeling the decision maker module of the system shown in Figure 1, which compares utility of additional observations with disutility of waiting. Other aspects of the proposed system are detailed in [1]-[3]. Disutility of waiting includes cost represented by the time required for additional testing and processing of the test results, the cost of observational, computational and datalink resources used to acquire additional observations, and the opportunity cost associated with delayed decision and action. The system to be described here maximizes the expected utility of delayed decisions while minimizing decision latency (time to decision). One of the major problems with using expected utility is that obtaining utility values from domain experts can be very costly and time consuming. There is also a consistency problem: different experts tend to provide different utility preferencing. Here we simplify the utilities-from-experts problem by employing reinforcement learning to learn utilities directly from environmental feedback signals.

Reinforcement learning has been successfully used for learning utilities in medical diagnosis systems [7] and we employ this approach in our work, with several notable modifications. First, we consider a decision deadline rather than assuming the process can continue until all the tests have been exhausted. Second, we deal with multiple hypotheses rather than with binary decision ("does a patient have this particular disease?"). Third, we assume that utilities decrease with each additional observations (increased opportunity cost), and, fourth, the decisions take into account not only a decision-maker's belief based on current observations but decision maker's cumulative-over-time belief up to and including the current observation. Belief combination is conducted within the framework of evidence theory. Fifth, the process of learning utilities is coupled with the process of learning the beliefs of the agents employed by the system.

2 Utility-based model of the sequential decision-maker

The task of the sequential decision-maker is to employ information obtained from the fusion center to choose whether to decide now (and if so which hypothesis), or defer decision and request another observation. If the action is decide now and a certain hypothesis is selected, the decision-maker presents the decision to the environment, which evaluates it, notes the decision time t, and returns a reinforcement signal used for adjusting decision functions of the system. Delayed reinforcement learning is used, since in most cases a decision will be made only after several observations. If action "defer decision" is chosen, agents will provide additional information based on new observations to FC, which will combine this information with information obtained at he previous step and transmit this updated information to the action selection unit (decision maker).

Let episode j be a subsequence of observations $\overline{X}_{T_{j-1}+\Delta t},\ldots,\overline{X}_{T_j}$ between two decision maker-

environment interactions at time T_{j-1} and T_j. Let N_j be the number of observations in episode j, i.e. $N_j = (T_j - T_{j-1})/\Delta t$ where Δt is a time interval between two subsequent observations. Consider episode j: $t_{n_j} \in [T_{j-1} + \Delta t \quad T_j]$, where $t_{n_j} = T_{j-1} + n_j \Delta t$ and $1 \le n_j \le N_j$ ($n_j = 1$ corresponds to an observation made at $T_{j-1} + \Delta t$ and $n_j = N_j$ to an observation made at time T_j). Let $T = T_{j-1} + n_d \Delta t$ be the deadline by which a decision must be made: $T_j \le T$, where $n_d \ge N_j$ is the maximum number observations permitted before the decision maker reaches the deadline. At each time t_{n_j} and for each hypothesis θ_i, the decision maker obtains $BetP^{t_{n_j}}(\theta_i)$, the cumulative pignistic probability of hypothesis θ_i, a function of the basic probability assignment representing a combination of all basic probability assignments produced by the fusion center in episode j up to and including time t_{n_j}. Given the pignistic probability vector $\overline{BetP}^{t_{n_j}}$ it remains to define how these values are used to arrive at a decision ("defer decision" or "select hypothesis i"). The decision is deferred if we expect to improve the outcome of the decision-making process by acquiring one or more additional observations while suffering the increased decision latency. There may be several criteria to consider when we decide whether we can improve our decision by deferring it. For example, we will tend to wait if we expect that the probability of selecting the right decision or the confidence of our decision will increase sufficiently [3,8]. However, utilization of this criterion does not allow us to explicitly include cost of time and resources into our decision-making. For this reason in the research reported here we design a decision policy based on the Maximum Expected Utility Principle [9]. The decision "wait for a new observation" or "decide on hypothesis θ_i" is based on the " value of information criterion" [9]. According to this criterion, a new observation is needed if the difference between maximum expected utility with the new observation and without the new observation is greater than cost of obtaining this new observation. The expected utility is assumed to be decreasing function of time since it includes the opportunity costs.

As it was mentioned in introduction, the core difficulty of utilizing the Maximum Expected Utility Principle is a problem of finding the utility for each decision. Instead of eliciting utilities from experts, a process that can be slow and expensive and produce subjective, possibly inconsistent utility values, we learn utility through interaction with the environment by utilizing a reinforcement signal on the quality of the decision made. There are several reinforcement methods developed for learning by delayed reinforcement [6]. One of them, the Temporal Difference method [6,10], has been proven to be successful for learning utilities [7]. Here we employ learning by Temporal Difference to converge towards an optimal policy to be used by the decision maker. The process of learning utilities is thereby coupled with the process of learning the beliefs of the agents employed by the system. In essence we learn how valuable each hypothesis is to us as we learn our beliefs that each hypothesis is true.

3 Learning method

Temporal difference methods (TD(λ) [10]) learn by employing the difference between temporally successive predictions. They have two major advantages over other prediction methods, they are more incremental and easier to compute; and they learn faster. TD(λ) uses a sequence of input data X_n, $n = 1,...,N$ to produce a sequence of estimates P_n, which are functions of corresponding inputs X_n and weights W_n. These estimates are predictions of the final reinforcement Z, which is defined as a data point N+1.

TD(λ) updates weights according to the following equation:

$$\Delta W_n = \alpha(P_{n+1} - P_n)\sum_{m=1}^{n} \lambda^{n-m} \nabla_w P_m . \qquad (1)$$

Here λ is a convergence parameter in the range [0 1]. In the online version of this algorithm:

$$W_{n+1} = W_n + \alpha(P_{n+1} - P_n)e_n, \quad e_{n+1} = \nabla_W P_{n+1} + \lambda e_n \qquad (2)$$

where e_n is the eligibility factor [6,10]. Let $U(t) = (u_{il}(t))$, $u_{il} \le 1$ be a time-dependent utility matrix where $u_{il}(t)$ is the utility of selecting hypothesis θ_i when the true state of nature is θ_l. We represent time-dependent utility as $u_{il} \cdot f(t)$ and $U(t) = (u_{il} \cdot f(t)) = U \cdot f(t)$, where $f(t)$ is a decreasing function of time with $f(1) = 1$ that guarantees that the observations will be stopped and a decision made before the deadline t=T. With each deferred decision we also consider a cost $C \in [0\ 1]$, which does not depend on time and represents the disutility of increased decision latency and the cost associated with resource utilization.

Given the initial observation for episode j, the agents determine their beliefs and FC combines them and computes $\overline{BetP}^{t_{n_j}}$ produced at time $t_{n_j} \in [T_{j-1} + \Delta t \quad T_j]$. We define the vector of expected utilities $\overline{E}(t_{n_j}) = U(t_{n_j}) \cdot \overline{BetP}^{t_{n_j}}$ where $E_i(t_{n_j})$ is the expected utility of selecting hypothesis θ_i. If we

consider the elements of the utility matrix as weights to be learned and $\overline{BetP}^{t_{n_j}}$ as inputs at time t_{n_j}, we can learn utility by changing the weights as in equation (2). Since the expected utility matrix is a linear function of $\overline{BetP}$, the utility matrix and eligibility functions are updated after each step as follows:

$$\begin{aligned} \Delta U_{t_{n_j}} &= \alpha U(t_{n_j})(\overline{BetP}^{t_{n_j+1}} - \overline{BetP}^{t_{n_j}})\bar{e}'_{t_{n_j}}, \\ \bar{e}_{t_{n_j+1}} &= \overline{BetP}^{t_{n_j}} + \lambda \bar{e}_{t_{n_j}} \end{aligned} \quad (3)$$

At the end of episode j (at time $T_j < T$) when the decision maker decides to stop observations and select a hypothesis θ_i, reinforcement z_{il} becomes known and utility u_{il} is updated by:

$$\Delta u_{il} = \alpha u_{il,T_j}(z_{il} - BetP_i^{T_j})e_{i,T_j}. \quad (4)$$

According to the value of information criterion decision time T_j is computed as the first time such that the difference between the maximum of expected utility at time T_j and the maximum expected utility at time $T_j - \Delta t$ is not larger than cost associated with additional observations, that is, decision time T_j corresponds to the first time satisfying

$$\max_i E_i(T_j) - \max_i E_i(T_j - \Delta t) \le C \quad (5)$$

There are two ways of defining the learning process. We can assume that the utility matrix we learn is time independent and represents the utility (cost) of selecting a particular hypothesis independent of the number of observations required to make a decision, $U = (u_{il})$. Then the factor of time becomes important only for selecting an action: ''defer decision '' or ''select hypothesis θ_i'' and T_j is computed as the first time such that:

$$\begin{aligned} &\max_i(U_{T_j}\,\overline{BetP}^{T_j} \cdot f(T_j)) \\ &- \max_i(U_{T_j} \cdot \overline{BetP}^{T_j-\Delta t} \cdot f(t^* - \Delta t)) \le C \end{aligned} \quad (6)$$

This approach is equivalent to eliciting expert utilities when experts are required to provide "absolute" time-independent utilities. In this case reinforcement is represented a time independent matrix $Z = (z_{il})$.. Alternatively, we can learn time-dependent utilities $(U = (u_{il} \cdot f(t))$ and T_j is computed according to (6). This case is equivalent to an expert knowledge elicitation process when it is assumed that expert knows and can explicitly determine how the utilities change with time as the deadline approaches. In this case we utilize time-dependent reinforcement. In many cases of practical interest time-dependent utilities are more natural models. For instance, the true utility of deciding hypothesis i decreases sharply as we cross that decision time beyond which there is insufficient time to react effectively to that correctly identified situation.

4 Case study

In order to evaluate the performance of the presented sequential decision-making model, we used a data set composed of 2545 forward looking infrared (FLIR) ship images from the U S Naval Airfare Center, China Lake, California. Images were digitized into 256x64 arrays of 8-bit pixels. Each ship image belongs to one of the eight classes listed in Table 2. These classes were further aggregated into two groups: friend and foe. All container ships were in group friend, all cruisers and frigates in group foe, and so on. Figures 2 and 3 present typical silhouettes for the 8 classes listed in Table 1.

The features used include seven moments given in [11]. These moments are invariant under translation, rotation and scale. But these moments deliver information primarily of the global shape of the object and represent poorly the details of the object. In [12] four features were added by fitting an auto regressive model to one-dimensional sequence of the projected image along the horizontal axis. The quality of each image varies considerably depending on the distance of the ship to the camera and the noise on the image.

In our experiments we divided all the features into three groups. Each group contained features with the lowest class-conditional correlation. The first group included the first, second and the seventh invariant moments, the second group included the remaining four invariant moments, and the third group contained the rest of the features. We conducted experiments with three agents, each of them using the specific group of features described above as input.

Testing was performed using cross-validation in which the group reserved for testing was fixed at 20% of the total available dataset. The training and test data were grouped into episodes of ten patterns drawn from the same object class. The object class was first selected with uniform probabilities over the eight classes, then the samples were chosen randomly from the portion of the test image set associated with that object class. The results shown in Figs. 4-7 were obtained after 40 iterations and were averaged over 5 runs.

There were several series of experiments conducted with each experiment corresponding to a different reinforcement matrix (Table 2). Each of the matrices represents a specific attitude of the decision-maker towards false alarm, correct binary recognition (friend/foe), and correct recognition of each ship class. Experiment 1 corresponds to the case in which the environment gives positive reinforcement (reward) in the case of correctly recognized classes, negative reinforcement (punishment) for a mistake in recognizing a group, and null reinforcement for correctly recognizing

a group but not the class within the group. Experiment 3 corresponds to the case in which equal reward is given for recognizing correct classes, and equal punishment for mistaking friend for foe (foe/friend) and vice-versa (friend/foe). Experiment 2 is similar to experiment 3 but the value of the reward is smaller. In experiment 4, reward is given only for correctly recognized classes, which is however, not equivalent to the case in which only classes are considered since negative reinforcement is given to incorrectly recognized groups not incorrectly recognized classes. In experiment 5 the highest reward is given to correctly recognized class, a smaller reword is given to correctly recognized group while punishment for mistakes "foe/friend" is more severe than punishment for mistake "friend/foe".

Table 1. Ship classes

Class Name	Class Number	Group Name	Number of Images
Destroyer	1	Foe	340
Container	2	Friend	455
Civilian Freighter	3	Friend	186
Auxiliary Oil Replenishment	4	Friend	490
Landing Assault Tanker	5	Foe	348
Frigate	6	Foe	279
Cruiser	7	Foe	239
Destroyer with Guided Missile	8	Foe	208

Each experiment comprised 2 variants:

- Learning time-independent matrix of utilities (matrix Z does not depend on time) *f(t)* is a decreasing function of time. In this case time was considered only for deciding on actions " wait" or "select a hypothesis". There were several different functions considered: $f(t) = 2 - \exp(1.025t)$ and $f(t) = \exp(-1.025t)$, and $f(t) = \exp(-0.2t)$. In this case reinforcement is as shown in table 2.
- Learning time-dependent utilities. Here *f(t)*, a decreasing function of time, was incorporated into the learning process. Reinforcement was also a function of time: $Z = (z_{ik}\phi(t))$, where (z_{ik}) is shown in Table 3, $\phi(t) = f(t)$ if $z_{ik} > 0$ and $\phi(t) = \exp(1.025t)$, or $\phi(t) = 2 - \exp(-1.025t)$, or $\phi(t) = 2 - \exp(0.2t)$ if $z_{ik} < 0$. Similar

In each case of each experiment we evaluated the performance of the system. We concentrated on the ability of the system to learn to maximize the utility matrix while interested in the correct group designation, the important binary distinction friend-foe. We were also interested in the more refined identification of ship class and the ability of the system to decrease the average number of observations in an episode (the average number of observations used by the system to arrive at a decision) as the result of learning.

Figures 4 -7 present results of some of those experiments. Here we considered

- the time function $f(t) = \exp(-0.2t)$
- Reinforcement time function $\phi(t) = \exp(-0.2t)$ if $z_{ik} > 0$ and $\phi(t) = 2 - \exp(-0.2t)$ otherwise.
- Reinforcement matrix (z_{ik}):

  $z_{ik} = 1$, if selected class i is the same as observed class k (i=k),

  $z_{ik} = 0.5$, if both selected class i and observed class k belong to the same group (friend or foe),

  $z_{ik} = -0.5$, if selected class belongs to the group "friend" and observed classes belong to the group "foe".

  $z_{ik} = -1$, if selected class belongs to the group "foe" and observed classes belong to the group "friend".

The performance of the system, in which agents and utility were trained by reinforcement in time-dependent as well as time independent manner is compared with the performance of the system, in which reinforcement was used for training agents only. The reinforcement matrix was used for utility initialization in both cases.

As we can see from Figures 4 -7, the performance of the system trained with time-dependent utility matrix is not much different than the performance of the system trained with a time-independent utility matrix. While the decision utility and recognition accuracy is slightly better in the time-dependent case, the number of observations required to achieve these result is somewhat higher than in the time-independent case. The results of both systems are superior to the performance of the system, in which the utility matrix remains constant. Decision utility obtained, as the result of training is higher and can be achieved with a smaller number of observations per episode. Although the accuracy of group recognition increases significantly as the result of utility training, the class recognition accuracy does not improved as much. This fact can be explained by the nature of reinforcement function, which emphasized group recognition and did not provide punishment for wrong class recognition.

Figure 2. Images of the 3 classes of ships (Friends)

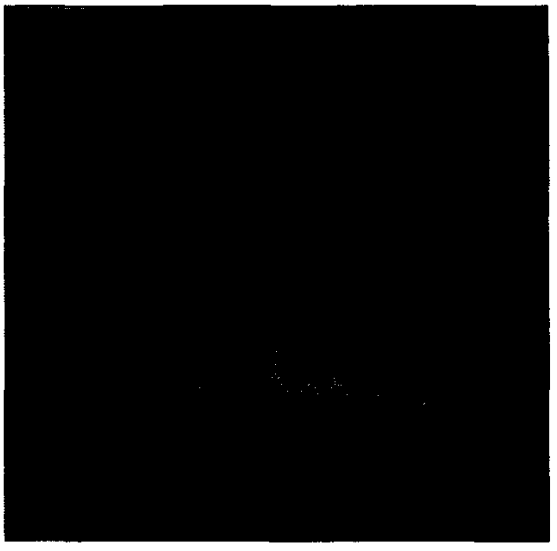

Figure 3. Images of 5 classes of ships (Foes)

Table 2. Time independent components of reinforcement matrixes.

	Reinforcement for correct class recognition	Reinforcement for correct group recognition		Reinforcement for mistake in group recognition	
		Friend/friend	Foe/Foe	Friend/Foe	Foe/friend
Experiment 1	1	0	0	-1	-1
Experiment 2	0.7	0.7	0.7	-1	-1
Experiment 3	1	1	1	-1	-1
Experiment 4	1	-1	-1	-1	-1
Experiment 5	1	0.5	0.5	-0.5	-1

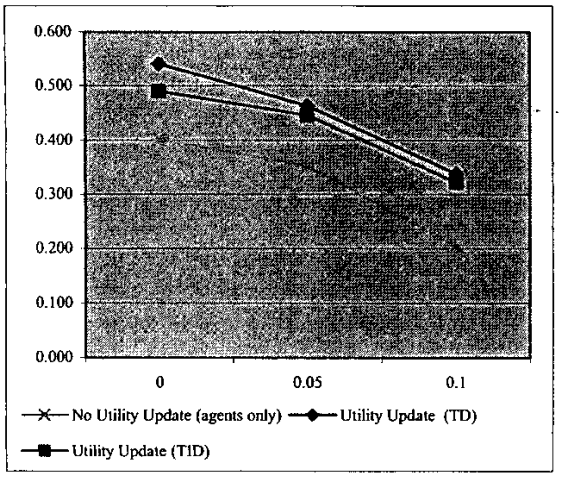

Figure 4. Utility as a function of cost

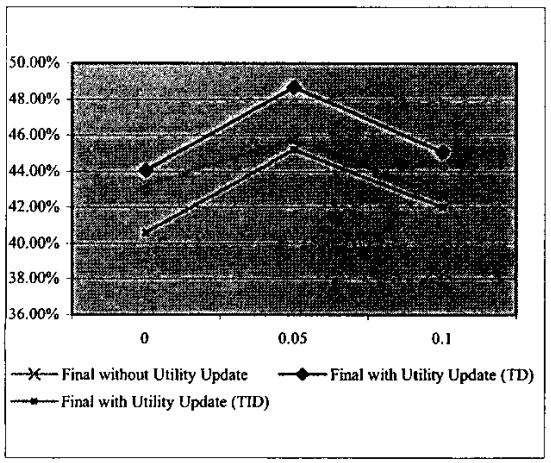

Figure 5. Correct group decisions (%)

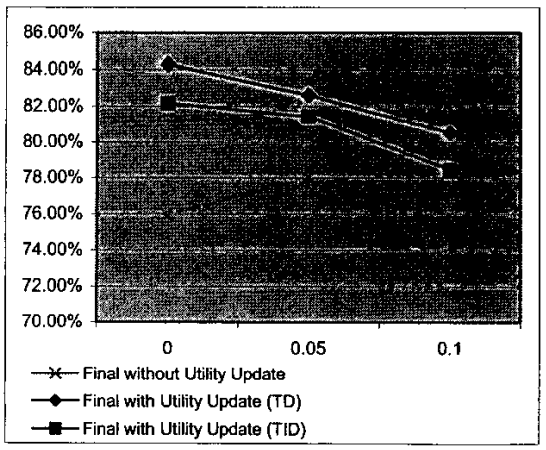

Figure 6. Correct class decision (%)

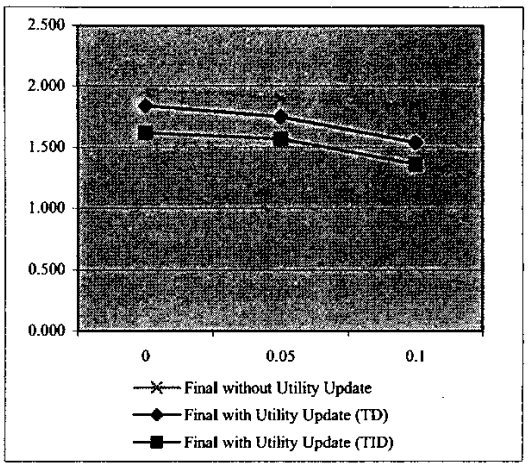

Figure 7. Average number of observations as a function of cost

5. Conclusion

This paper addresses several issues, which arise in designing a homogeneous non-communicating multi-agent system guided by reinforcement learning. In particular a process of sequential decision-making with limited resources in time-constrained situations has been investigated. A new approach to building a utility-based model of cost-sensitive decision-making in time-constrained situation has also been developed. A modification of the method suggested by Stesmo and Sejnowski [7] for medical decision-making has been introduced. This modification includes utilization of evidence theory, multiple hypotheses rather than binary hypothesis, time-varying utility, and exploitation of all the beliefs produced by the fusion center in each episode up to and including time when decision is made. In addition, the process of learning utilities is coupled with the process of learning the beliefs of the agents employed by the system.

The case study reported shows the feasibility and benefits of employing a temporal difference model in the context of evidence theory for sequential decision-making. The case study demonstrates that decision utilities as well as the system classification ability can be improved through reinforcement learning of the utility matrix. It shows that both time-dependent and time-independent utility matrices can be learned from feedback on the quality of the decisions rather than directly elicited from an expert, which simplifies the slow, expensive and potentially error-prone expert knowledge elicitation process.

The presented methods are not problem specific and can successfully be used for both military and non-

military applications. They can be used, for example, in multi-sensor moving target recognition, in single-sensor classifiers in which the feature set is partitioned across several distinct computing agents and parallel-processed to reduce decision latency, in the situation assessment problem requiring selection of a certain hypothesis about the state of the environment, for building a reinforcement learning-based sensor management algorithms, or to improve medical decision-making in life-threatening situations.

Additional experiments are necessary to determine the relative performance of these two homogeneous non-communicating multi-agent systems with statistical reliability, and to investigate the problem of incorporating agents' reliability into the sequential decision-making process. More research is also needed in order to address fundamental issues of the problem of distributed learning in problem solving systems such as utilization of *a priori* knowledge, incorporation of symbolic and numeric information, convergence of the process, etc.

6. Acknowledgements

This research was supported by the U.S. Air Force Office for Scientific Research (AFOSR) under Contract No.F49620-01-1-0371, whose support and support of Jeffery Layne, Stan Musick, and Raj Malhotra from the Air Force Research Lab (AFRL, Sensor Directorate) are gratefully acknowledged.

The infrared (FLIR) ship images from the United States Naval Airfare Center, China Lake, California. were provided by Dr. Jack Sklansky of the University of California at Irvine and obtained from Dr. Pierre Valin of Lockheed Martin, Canada.

7. References

[1] Rogova, G, Menon, R, *Decision fusion for learning in pattern recognition*, in *Proc. of the FUSION'98 - First Conference on Multisource- Multisensor Information Fusion,* 1998, 191-198.

[2] G. Rogova, J. Kasturi, Reinforcement Learning Neural Network For Distributed Decision Making, in *Proc. of the FUSION'2001-Forth Conference on Multisource-Multisensor Information Fusion,* 2001, Montreal, Canada

[3] G. Rogova, P. Scott, C. Lollett, Distributed Reinforcement Learning For Sequential Decision Making, In: *Proc. of the FUSION'2002-Fifth Conference on Multisource- Multisensor Information Fusion,* July 2002.

[4] Shafer, G., *A Mathematical Theory of Evidence*, Princeton, MIT Press, 1976.

[5] P. Smets and R. Kennes, The transferable belief model, *Artificial Intelligence*, vol. 66,1994, 191-243.

[6] R. S. Sutton and A. Barto, *Reinforcement Learning: An Introduction*, MIT Press 1998.

[7] M. Stesmo, T. Sejnowski, Using Temporal-Difference Reinforcement Learning to Improve Decision-Theoretic Utilities for Diagnosis, In: *Proc. 2nd Joint Symposium on Neural Computation, University of California, San Diego and California Institute of Technology*, June 1995.

[8] Baum, C.W., Veeravilli V.V., A sequential procedure for multihypothesis testing, *IEEE Transactions Information Theory* Vol. IT-40, 1994-2007 November 1994.

[9] J. von Neuman, O. Morgenstern, *Theory of Games and Economic Behavior*. Princeton University Press, Princeton, NJ, 1947.

[10] Sutton, R. S. (1988). Learning to predict by the method of temporal differences. *Machine Learning*, 3, 9–44.

[11] Park, Youngtae and Jack Sklansky, Automated Design of Linear Tree Classifiers. *Pattern Recognition,* Vol. 23, No 12, pp. 1393-1412,1990

[12] M. K. Hu, Visual pattern recognition by moment invariants, *IRE Transactions on Information Theory*, No. 8, 179-187(1962)