SIMULATION OF A COGNITIVE ALGORITHM FOR A DISTRIBUTED ROBOTIC SENSING NETWORK[1]

David Schoenwald, Patrick Xavier, Edward Thomas, Chris Forsythe, Eric Parker
Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185-1004

ABSTRACT

This paper presents simulation and analysis of a collective of autonomous unmanned ground-based vehicles navigating a building searching for increasing smoke concentrations. The vehicles communicate smoke concentrations to each other to determine the location of the highest concentration value. The data generated from the robots' sensors is used to activate a semantic network to generate data for further cognitive operations. Statistical analysis is employed on the data to identify schema and themes, which enable the robots to convey a story of their experiences, thus emulating human episodic memory.

KEYWORDS: modeling & simulation, autonomous robotic vehicles, cognitive algorithms, distributed network, human cognitive emulation

INTRODUCTION

This work examines the use of computer simulation of a collective of embodied agents to generate episodic memory for use in human cognitive emulation. Ground-based robotic vehicles attempt to search a building for smoke and place a robot at the highest smoke concentration the collective has found. The robotic collective generates input data for a human cognitive emulation technique whereby cognitive models associated with each robotic entity enable a "meaningful" representation of experiences that mimics human episodic memory.

The vehicles communicate with each other via a token ring network to inform each other as to their position and current highest smoke levels. The smoke is randomly generated for each run of the simulation, and the robots follow increasing smoke densities while attempting collective wall-following-based exploration through the corridors. Collision avoidance has the highest priority, followed by wall-following to prevent getting disoriented. The robots' sensors activate a semantic network to generate the data for subsequent cognitive operations. Statistical analysis of data obtained through casually sampled simulation runs allowed derivation of schema and themes based on activation of the semantic network. Pattern recognition techniques, including neural networks, are being developed to map a robot's experience to these schema and themes. The goal is for the robots to be able to tell a story of their experiences based on these schema and themes, emulating human episodic memory.

BACKGROUND

Modeling and simulation for effects-based operations introduces substantially greater demands for realism in synthetic entities than has been typical. To realistically model human behavior associated with effects-based operations, it is asserted that behavioral models for synthetic entities must incorporate the following attributes:

[1] This work was performed at Sandia National Laboratories. Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

- emotional processes including the interaction between emotions, and cognitive processes and arousal mechanisms
- representations of knowledge that provide a broad range of relevant and also irrelevant behavioral responses
- mechanisms to address variations in knowledge and emotional associations attributable to cultural differences
- mechanisms that enable non-linear patterns of response and the capacity to respond appropriately to non-linear responses

In addition, tools must also provide the ability to model social behavior providing a framework wherein numerous, individually unique, synthetic human agents may interact with emergent behavior arising from the collective.

Sandia National Laboratories has undertaken a program of research and development to develop realistic computational models of human cognitive and psychological processes[1]. This work has led to a technology referred to as "Human Cognitive Emulation." Emulators are being explored for a variety of applications that include synthetic humans for simulation environments, control processes for intelligent machines, and agent representations for technological solutions to augmented cognition. Initial emphasis in the development of human emulation has focused on the computational representation of naturalistic decision making with particular attention to a recognition-primed model[2]. The decision maker is attributed knowledge of "situations," or "schema." These situations represent familiar contexts and are meaningful in two respects. First, there are likely co-occurrences that provide the basis for expectations. For example, in a restaurant situation, if handed several pages bound together, your initial perception will likely be that you have been handed a menu. Second, there are goal-action sequences. Implicit to recognition of a situation, there is recognition of goals, or attainable states, and the actions needed to realize those goals, including likely intermediate states. For example, in a close quarters battle, tactics provide the situations and with each tactic, there are goal-action sequences, accompanied by appropriate roles and task responsibilities

SIMULATION MODEL

The multiple-vehicle problem in planar space is essentially a generalization of one-dimensional analysis[3]. But when obstacles such as walls and other vehicles as well as the need for communication between vehicles are taken into account, the ability to analytically solve the problem becomes very difficult. Thus, we implemented a study of multiple vehicles under such constraints in a simulation environment developed at Sandia National Laboratories called Umbra[4]. Umbra enables the simulation of multiple autonomous agents with a variety of physical phenomena such as RF (radio frequency) communications, interactions with solid objects (e.g. collisions), ultrasound communication, IR (infrared) detection of objects, vehicle physics, terrain descriptions, and other phenomena. All of these physical attributes can be simulated simultaneously with a graphical visualization that allows the monitoring of the vehicles' performance over the terrain.

Such a simulation was implemented for the case of multiple, small, wheeled vehicles traversing a single floor in a building with multiple corridors, rooms, and entrances. The vehicles are models of actual hardware. Each vehicle contains 4 IR sensors for detecting objects between 0.15m and 0.46m on its four sides (see Figure 1). The vehicles also contain RF communication devices to be able to converse with other vehicles within a 30m line of sight (LOS) or roughly 10m through walls. They also have ultrasound capability to measure the distance between them provided they are within 10m of each other and in LOS (more details can be found on this in [3]). The vehicle physics models are simple and proved adequate on a smooth surface. The building model was generated as a CAD model and contains several connected hallways as well as a multitude of variable size rooms. The control algorithms for the vehicles must avoid contact with walls and other vehicles. Beyond that, the control algorithms enable the collective to place a member at the maximum smoke concentration found in the building. Note that a strict mathematical model of this situation is intractable. This is due to both the discrete event-based nature of the communications as well as the dynamic physics models with very complicated interactions between the vehicles and obstacles.

Thus, the simulation shows stability in a qualitative rather than strictly mathematical fashion. However, future work will focus on demonstrating that these control algorithms are robust to modeling uncertainty.

The restriction that vehicles can't move through walls, doors, or each other essentially ensures they remain inside the building. This is accomplished via rules that use the IR sensors to follow walls down a hallway. This enables the vehicles to move throughout the building, though not necessarily in any prescribed fashion. Further restrictions on the vehicles involve the maintenance of a continuous RF communication network requiring that vehicles stay within 30m of each other or less if LOS is lost (i.e. they may have to stay at a wall junction to maintain LOS).

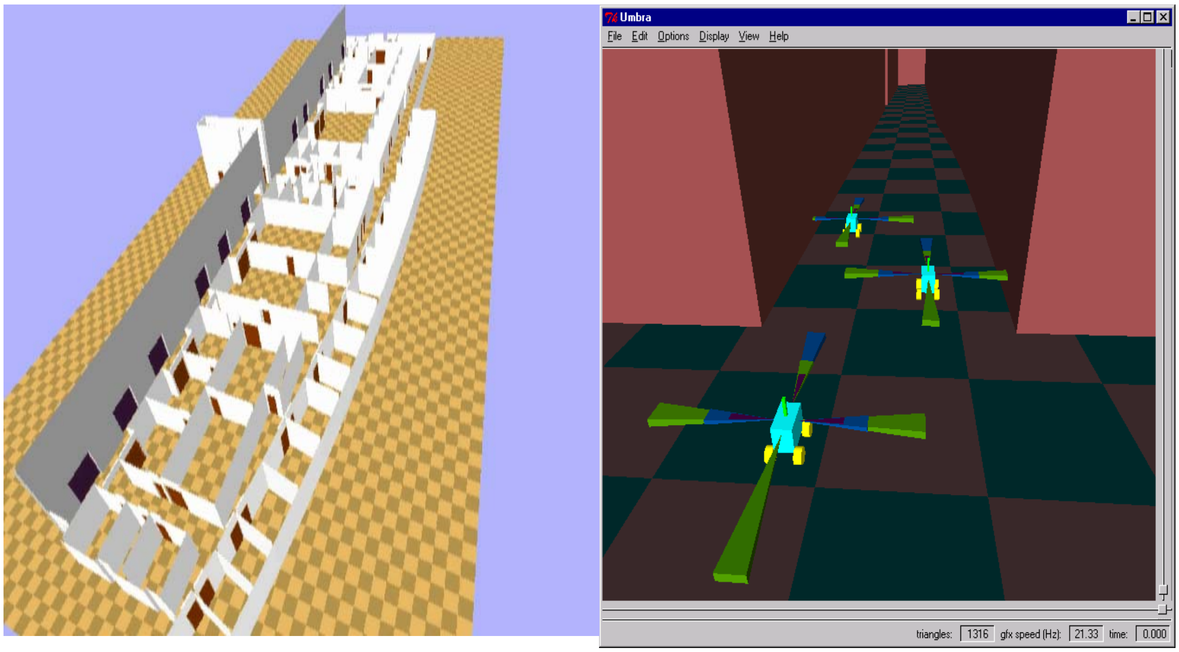

Figure 1. Detailed simulation of multiple vehicles navigating a building. The left view shows a cut-away view of the building under study. The right view shows a close-up of vehicles with their IR sensors visible.

SCHEMA IDENTIFICATION

A total of 20 simulation runs, each involving eight robots, were conducted within the framework of Umbra. The source of smoke varied from simulation to simulation so as to induce different behavior across runs. For these simulations, the initial location/status of the eight robots was constant across runs. Data was gathered from these runs to identify schema and themes via which the robots would report their experiences searching for smoke in the building.

A complex, multi-dimensional data vector (with binary and continuous dimensions) is used to define the status of each robot at any point in time during a simulation. The status of each robot was sampled once per second over the duration of each 300-second simulation run. With 300 observations per robot per simulation run, the total data set consists of 48000=300*20*8 observations, where each observation consists of the 15 dimensions listed in Appendix 1. These 15 dimensions were chosen out of a set of 32 to keep the data set computationally tractable.

The analysis consists of several distinct steps. First, using a representative training set, *cluster analysis* was used to group the collection of observations into subsets or *clusters*. Clusters are interpreted using a *classification tree* model. All observations, over all simulations/robots, are partitioned by the classification tree rules into interpretable robot states. At this point each observation has been mapped from the complex, multi-dimensional data vector into a discrete state-space with relatively few states. This dimension reduction facilitates the analysis of temporal patterns exhibited by individual robots as well as the system of robots ans simplifies studying the differences in behavior from robot to robot and across simulation runs.

Cluster analysis is a form of *unsupervised learning* where the goal is to partition a collection of observations into subsets (or clusters) such that those observations within a cluster are more closely related to one another than observations assigned to different clusters[5]. The nature of unsupervised learning is that there is no knowledge of the true data structure. Two clustering algorithms are used: K-means clustering and DIANA.

The K-means algorithm requires initial estimates of the number of clusters and location of each cluster's multidimensional center, then iterates the following steps until convergence.

1. For each observation identify the closest cluster center in Euclidean distance.
2. Replace each cluster center with the average of all points that are closest to it.

Convergence is declared when the cluster assignments do not change. The K-means algorithm initialized a number of times, each time with a different specification for the number of clusters.

At convergence for each case, the total within-cluster variability is used as a measure to select the number of clusters. The goal is to obtain a partitioning that involving few clusters so that the level of within-cluster variability is acceptably small. Another goal is to develop a set of clusters such that the number of observations per cluster is not too small.

DIANA is a clustering algorithm (see [5]) that, unlike the K-means algorithm, is hierarchical in nature. That is, clusters at each level of the hierarchy are defined by combining clusters at the next lowest level. Classification tree modeling (see [6]) is a form of *supervised learning* where the objective is to partition the predictor variable space into regions that are homogeneous with respect to *known* classifications.

The data set of 48000 observations is too large to feasibly compute the cluster analysis and classification tree modeling. Therefore, the cluster analysis and classification tree modeling is based on a representative training set consisting of 800 observations. The training set is obtained by randomly selecting ten observations per robot per each of the first ten simulation runs. Thus, we have representation across all robots and simulation runs.

For both the K-means and DIANA algorithms a range of values from 1-10 is considered for the number of clusters. In the case of K-means, 5 clusters appear to provide an adequate partitioning of the training set. In the case of DIANA, 6 clusters provide a reasonable partitioning of the training set.

A classification tree analysis using an SPLUS implementation is performed using the sets of cluster associations developed by the K-means and DIANA algorithms. Interpretation of the tree structures result in 5 terminal nodes (states) in the case of the K-means algorithm and 6 states in the case of the DIANA algorithm (see Figures 2 and 3). The tree structures can be interpreted as follows.

In the case of the classification tree derived from K-means clustering, the primary partitioning of data is with regard to dimension-15 which is an indicator of whether the robot was or was not stopped. In particular, observations with a value of less than 0.8 for dimension-15 were passed to the left side of the tree and to the right side otherwise. State-5 is associated with a "moving robot" *and* a small value for dimension-4. That is, state-5 relates to a robot moving slowly in the x-direction. State-3 is associated with a robot that is moving relatively quickly in the x-direction. State-1, state-2, and state-4 are associated with robots that have stopped or nearly have stopped. The difference between state-1 and {state-2, state-4} is due to dimension-8, an indicator of how close the robot's current smoke level is to its previous maximum smoke level. Thus, state-1 pertains to robots that have stopped at a position where the smoke level is not close to the maximum smoke level that had previously been experienced by that robot. The difference between state-2 and state-4 is the presence/absence of an RF_Ping (dimension-14). For example, state-2 is associated with robots that have stopped *and* are at a position where the smoke level is close to the maximum *and* are not pinging.

In the case of the classification tree derived from DIANA clustering, the dimensions that lead to the definition of the state space are: dimension-13 (RF_Hear_Beacon), dimension-6 (current smoke level), dimension-4 (level of movement in x-direction), and dimension-5 (level of movement in y-direction). For example, state-1 is associated with robots that are not currently hearing a strong beacon signal *and* are detecting relatively low levels of smoke. Also, for example, state-4 is associated with robots that are hearing a strong beacon signal *and* moving quickly in both the x- and y-directions.

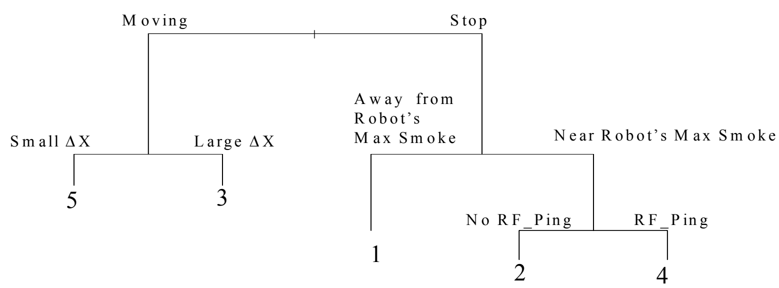

Figure 2- Classification Tree Derived From K-Means Clustering

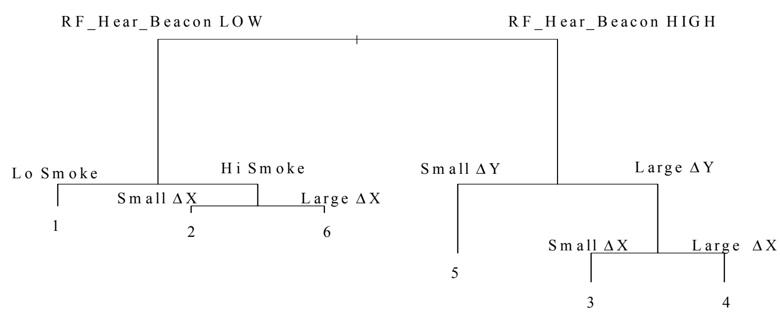

Figure 3- Classification Tree Derived From DIANA Clustering

DATA ANALYSIS – ANALYSIS OF STATE SPACES

The classification trees are developed using a training set of 800 observations. The partitioning rules associated with these trees are applied to the complete set of 48,000 observations. Thus, each of the 48,000 observations are assigned to a particular state: states 1-5 in the case of trees developed from the K-means clustering and states 1-6 in the case of the DIANA clustering. In most cases, robot-1 remains in state-2 during the next epoch. That is in about 95 % of all instances with robot-1, $S_{t+1} = 2$ given that $S_t = 2$. In other instances $S_{t+1} = 4$. Thus, the interpretation is that this robot is always *stopped* and occasionally *pinging*.

The different state space representations (K-means and DIANA) can be integrated to investigate the behavior of individual robots. Based on the analysis associated with K-means clustering, robot-1 clearly stands out by virtue of the fact that it resides exclusively in state-K2 and state-K4 (the *K*-prefix denotes a state associated with the K-means tree). Based on the analysis associated with DIANA clustering, robot-1 is also found to exhibit unusual behavior as it resides entirely in state D5 (the *D*-prefix denotes a state associated with the DIANA tree). Additional comparisons indicate five classes of robots with regard to their behavior: robot 1, robot 7, robot 8, robots 2&3, and robots 4,5,&6 (One might argue that robot 7 belongs with robots 4,5,&6).

One might summarize the different robot behaviors as follows. Robot-1 is the least mobile robot. It is always *stopped*, always hears a strong beacon signal, and is occasionally *pinging*. Robots 2&3 are usually stopped or moving slowly, near high levels of smoke, and are not hearing a strong beacon signal (states K1, K2, and D2). Perhaps robots 2&3 lead the way in exploring for the source of smoke. Robots 4,5&6 spend their time in a variety of states, most frequently D2, D3, and K5. Robot 7 behaves similarly to robots 4,5&6. However, robot 7 spends a larger proportion of time in state D2, which can be viewed as a *terminal* state. That is, once a robot enters this state, it is unlikely to leave it. Note that entry to D2 is exclusively through D6. Robot 8 is somewhat similar to robots 4,5&6 and robot-7. A notable difference is that robot 8 did not transition from K2 to K3 resulting in significantly fewer visits to K3 than robots 4,5&6.

CONCLUSIONS

This paper demonstrates the ability to identify behavior via schema that were developed without specific subject matter knowledge. While rather casual data selection enables identification of high-level behavior, more rigorous approaches are expected to yield more detailed information that can be applied to enhance system performance and application. Further data analysis might identify system behavior resulting in more rapid detection of the smoke's source or enable an assessment of whether fire or smoke location influences robot behavior.

APPENDIX – LIST OF STATUS DIMENSIONS

Dimension	Description
1	Time elapsed following start of simulation
2	Current *X*-coordinate: X_t
3	Current *Y*-coordinate: Y_t
4	Absolute value of change in *X*-coordinate (since previous epoch): X_t - X_{t-1}
5	Absolute value of change in *Y*-coordinate (since previous epoch): Y_t - Y_{t-1}
6	Current smoke level: S_t
7	Change in smoke level since previous epoch: S_t - S_{t-1}
8	Current smoke relative to maximum smoke since start: S_t - max{ S_1 , S_2 , ..., S_t}
9	Current smoke relative to global (over all robots) maximum smoke since start
10	IS_Beacon: binary variable that indicates whether or not robot is a beacon
11	IS_Last: binary variable that indicates whether or not robot is last
12	IS_Rover: binary variable that indicates whether or not robot is a rover
13	RF_Hear_Beacon: binary variable that indicates whether or not robot can hear a beacon
14	RF_Ping: binary variable that indicates whether or not robot is pinging
15	STOP: binary variable that indicates whether or not robot is STOPed

REFERENCES

[1] Forsythe, C. (2001). Conceptual Framework for a Comprehensive Computational Representation of Human Decision Processes. SAND 2001-1844, Sandia National Laboratories, Albuquerque, NM.

[2] Klein, G. (1997). An overview of naturalistic decision making applications. In C.E. Zsambok & G. Klein *Naturalistic Decision Making*, Mowah, NJ: Lawrence Earlbaum, 49-59.

[3] D. A. Schoenwald, J. T.Feddema, F. J.Oppel, "Decentralized Control of a Collective of Autonomous Robotic Vehicles", Proceedings of the 2001 American Control Conference, Crystal City, VA.

[4] E. J. Gottlieb, R. W. Harrigan, M. J. McDonald, F. J. Oppel, P. G. Xavier, "The Umbra Simulation Framework," Sandia Report: SAND2001-1533, June 2001.

[5]. *An Introduction to Cluster Analysis*, L. Kaufman, P. J. Rousseeuw, Wiley, 1990.

[6] *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Hastie, T., Tibshirani, R., Friedman, J., 2001, Springer-Verlag, New York, pp. 270-279