Experimental State Splitting for Transfer Learning

Clayton T. Morrison CLAYTON@ISI.EDU
Yu-Han Chang YCHANG@ISI.EDU
Paul R. Cohen COHEN@ISI.EDU
Joshua Moody MOODY@ISI.EDU
Information Sciences Institute, University of Southern California, 4676 Admiralty Way, Marina del Rey, CA 90292

Abstract

Jean is a model of early cognitive development based loosely on Piaget's theory of sensori-motor and pre-operational thought (Piaget, 1954). Like an infant, Jean repeatedly executes schemas, gradually extending its schemas to accommodate new experiences. We model this process of accommodation with the *Experimental State Splitting* algorithm. We present the algorithm and demonstrate, in three transfer learning experiments, Jean's ability to transfer learned schemas to new situations in a real time strategy military simulator.

1. Introduction

Jean is both a synthesis of ideas about cognitive development and the foundations of concepts, and an integrated software system that implements perception, action, learning and memory (Chang et al., 2006). From Piaget we borrow the ideas that children learn some of what they know by repeatedly executing schemas, and executing schemas is in a sense rewarding, and some new schemas are modifications or amalgamations of old ones (Piaget, 1954). The Image Schema theorists (Lakoff, 1987; Johnson, 1987; Mandler, 2004; Oakley, 2006) promote the ideas that primitive schemas are encodings or redescriptions of sensorimotor information; and these schemas are semantically rich, general, and extend or transfer to new situations, some of which have no salient sensorimotor aspects. Another idea, represented by various authors, is that semantic distinctions sometimes depend on dynamics — how things change over time — and so schemas should have a dynamical aspect (Thelen & Smith, 1994; Cohen, 1998; Talmy, 2003). These insights inspired our development of the Image Schema Language for representing primitive relations, environment dynamics, and actions (St. Amant et al., 2006). The experimental state splitting (ESS) algorithm provides a method for constructing new composite schema representations using the image schema language. Although ESS can learn policies for new situations from scratch, we are much more interested in how previously learned policies can accommodate or *transfer* to new situations. We present the ESS algorithm and results of its performance in three experiments specifically designed to measure the effects of knowledge transfer from previous learning in one scenario to another, different scenario. In the next section we introduce the ESS algorithm. We then describe the experiments, the transfer testing protocol and the results of the experiments.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

2. Learning: Experimental State Splitting

The basic idea of ESS is to grow state machines by incrementally elaborating state descriptions that make distinctions that previously were elided. These new distinctions introduce two new states where previously there was one. In order to make new distinctions, the ESS algorithm requires some criterion or measure to determine which splits are appropriate or worth introducing. For a general developmental account we want a general measure, not a task-specific one. To accord with the idea that learning is itself rewarding, this measure might have something to do with the informativeness or novelty or predictability of states. In Jean, the ESS algorithm uses a measure we call *boundary entropy*, which is the entropy of the distribution of next states Jean might transition into given the current state and an intended action. ESS cal-

culates the entropy of this distribution and uses it as a state splitting criterion: Jean is driven by ESS to modify its world model by augmenting existing states with new states that *reduce* the boundary entropies of state-action pairs. This augmentation is achieved by splitting an old state into two (or more) new states based on distinguishing characteristics.

More formally, ESS works as follows. We assume that Jean receives a set of schema features $F^t = \{f_1, \ldots, f_n\}$ from the environment at every time tick t; these features could be schema slots that represent sensor readings, for example. We also assume that Jean is initialized with a goal state s_g and a non-goal state s_0. S_t is the entire state space at time t. A is the set of all actions, and $A(s) \subset A$ are the actions that are valid for state $s \in S$. Typically $A(s)$ should be much smaller than A. $H(s_i, a_j)$ is the boundary entropy of a state-action pair (s_i, a_j), where the next observation is one of the states in S_t. A small boundary entropy corresponds to a situation where executing action a_j from state s_i is highly predictive of the next observed state. Finally, $p(s_i, a_j, s_k)$ is the probability that taking action a_j from state s_i will lead to state s_k.

For simplicity, we will focus on the version of ESS that only splits states; an alternative version of ESS is also capable of splitting actions and learning specializations of parametrized actions. The ESS algorithm follows:

1. Initialize state space with two states, $S_0 = \{s_0, s_g\}$.
2. While ϵ-optimal policy not found:
   a. Gather experience for some time interval τ to estimate the transition probabilities $p(s_i, a_j, s_k)$.
   b. Find a schema feature $f \in F^i$, a threshold $\theta \in \Theta$, and a state $s_i \in S$ to split that maximizes the boundary entropy score reduction of the split:

$$\max_{S,A,F,\Theta} \left[H(s_i, a_i) - \min(H(s_{k_1}, a_i), H(s_{k_2}, a_i)) \right]$$

      where s_{k_1} and s_{k_2} result from splitting s_i using feature f and threshold θ: $s_{k_1} = \{s \in s_i | f < \theta\}$ and $s_{k_2} = \{s \in s_i | f \geq \theta\}$.
   c. Split $s_i \in S_t$ into s_{k_1} and s_{k_2}, and replace s_i with new states in S_{t+1}.
   d. Re-solve for optimal plan according to p and S_{t+1}

The splitting procedure iterates through all state-action pairs, all of the schema features F, and all possible thresholds in Θ and tests each such potential split by calculating the reduction in boundary entropy that results from that split. This is clearly an expensive procedure. We are currently investigating methods to speed up the search for splits with heuristics that limit Jean's attention to relevant features and state-action pairs.

3. Experiments

We tested Jean's transfer of schemas between scenarios in the 3-D real time strategy game platform **ISIS**. In each scenario in the experiments, Jean controlled a single squad at the squad level with another, smaller but faster squad controlled by an automated but non-learning opponent. In all of the scenarios, Jean's mission was to command its units to engage and eliminate the opponent force. The knowledge Jean acquired and transfered between scenarios involved learning policies for choosing among the actions of *run*, *crawl*, *move-lateral*, and *stop-and-fire*, respecting the engagement ranges, possible entrenchment of the opponent, and some terrain features (mountains).

3.1. Scenarios

All scenarios were governed by a model of "engagement ranges" that determined how the squads may interact and how the opponent controller would respond to Jean's actions. Engagement ranges were defined in terms of the distance between Jean's force and the opponent. At the *Outer Range* (beyond 250 meters), as long as the opponent is within line of sight (i.e., not obscured by terrain features), Jean can locate the opponent force but the opponent cannot visually contact Jean's force. Within 250 meters, the opponent can see Jean's force (make *Visual Contact*) unless Jean's forces are crawling (and haven't yet fired).Within 200 meters, *Firing Range*, either force can fire on the other. Finally, within 100 meters the two forces have reached *Full Contact* and even if Jean's forces are crawling, they will be sighted by the opponent.

Measuring transfer is both a problem of theory and methodology. In the work we report here we adopted the **B/AB** protocol to measure how learning in one condition, A, transfers to learning and performance in condition B. The protocol is simple: measure performance over time while Jean is exposed to condition B (the B-alone condition). Then compare this performance with the performance when Jean is exposed to condition B *after* first being trained on A (the AB condition). Each experiment thus defines an A scenario

and a B scenario, and the learning curves generated by measuring performance over time in the B-alone condition and the B performance in the AB condition were compared.

The three experiments to test transfer of knowledge between pairs of the scenarios were defined as follows:

- *Experiment 1*: Condition A consisted of training Jean to catch and attrit the opponent in an open field and always starting within 250 meters. Condition B consisted of the same open field, except 50% of the time Jean started within 250 meters, the other 50% of the time beyond 260 meters – this was referred to as the *full open field* scenario.
- *Experiment 2*: This time, condition A had Jean always start beyond 260 meters. Condition B was again the same as condition B in Experiment 1: the full open field.
- *Experiment 3*: Condition A consisted of training in the full open field scenario. Condition B was just like the full open field except that now there was a single hill in the middle of the field that could obscure Jean's view of the opponent. Now Jean had to learn to find the opponent visually before utilizing any strategy from condition A.

3.2. Results

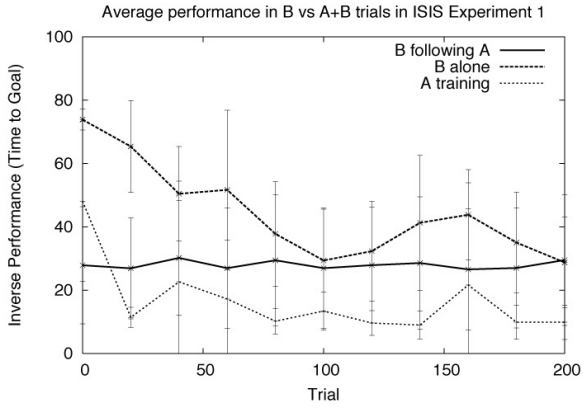

Figure 1. Experiment 1 graph showing learning curves for the A, AB, and B conditions, averaged over eight replications of the experiment. One standard deviation is shown with the error bars. Each point of each curve is the average of ten fixed test trials. These test trials are conducted every 20 training trials. The x-axis plots the number of training trials that the agent has completed.

The learning curves for the B and AB conditions in Experiment 1 are shown in Figure 1. The graph also shows the learning curve for the A condition, merely to verify whether Jean has indeed successfully learned a good policy for the A condition. For example, it is clear in Figure 1 that Jean learned a good policy in the A condition, where the enemy units are initialized close to Jean's position. Experiment 1 also shows that Jean receives a significant benefit when it moves to the B condition after training in the A condition. The AB learning curve starts out immediately with much better performance than the B learning curve, and after 200 training trials, the two curves reach approximately the same level of performance.

It is clear that in Experiment 1, Jean succeeds in transferring knowledge from the A condition to help her perform better in the B condition immediately. To summarize this improvement, we use the notion of a

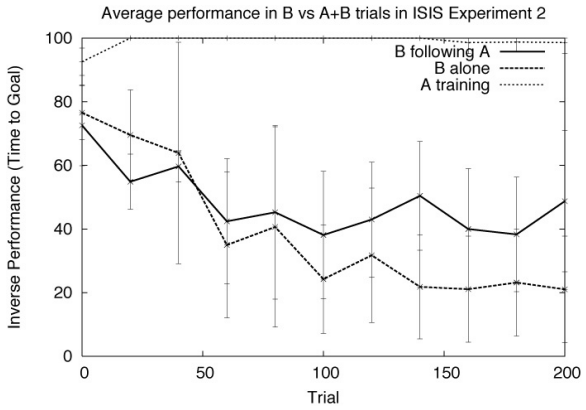

Figure 2. Experiment 2 graph, again showing learning curves for the A, AB, and B conditions, averaged over eight replications of the experiment. The x-axis plots the number of training trials that the agent has completed.

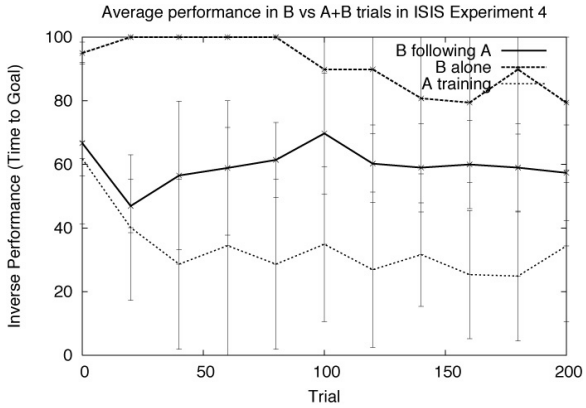

Figure 3. Experiment 3 graph, again showing learning curves for the A, AB, and B conditions, averaged over eight replications of the experiment. The x-axis plots the number of training trials that the agent has completed.

	Performance ratio r	P-value	2.5% quantile	97.5% quantile
Experiment 1	1.591	0.035	1.02	2.97
Experiment 2	0.970	0.557	0.63	1.34
Experiment 3	1.531	0.0034	1.24	1.85

Table 1. Table showing the transfer ratio of the AB and B cases in the various experiments, along with associated p-values and confidence intervals.

transfer ratio

$$r(B, AB) = \frac{Area(\overline{B})}{Area(\overline{AB})},$$

where B and AB denote the set of n learning curves in the B and AB conditions respectively, $\overline{X}$ is the mean learning curve for a set of learning curves X, and $Area(X)$ is the area under learning curve X. A large ratio indicates better transfer. In Experiment 1, the transfer ratio r is 1.591, and the benefit is significant at the 3.5% level (i.e. our p-value is 0.035). The p-value is calculated using the randomization-bootstrap method (Cohen, 1995), and measures the fraction of the sampling distribution for the null hypothesis in which r is greater than our observed data.

In Experiment 2, it is equally clear that Jean does not succeed in achieving any transfer. On closer inspection, we realize that Jean does not ever learn anything useful in the A condition of Experiment 2. This can be seen from Figure 2, where the learning curve for the A condition stays almost at 100 throughout the training trials. This is due to the difficulty of the scenario. Jean is always initialized far away from the enemy units, and must learn a policy for killing them by exploring a continuous, high-dimensional feature space using her four available actions. Many of these actions result in the enemy soldiers detecting Jean's presence and running away, thus reducing Jean's chances of ever reaching her goal by simple exploration. Since Jean does not learn anything useful in the A condition, we can expect that her performance in the AB case will not be any better than in the B case, and indeed, we can see that this is true in the observed data. Our measure r is approximately one, indicating little difference between $\overline{B}$ and $\overline{AB}$, and the p-value is 0.557, as we would expect if there is indeed no difference between the B and AB conditions.

Finally, in Experiment 3, Jean appears to transfer her learned knowledge from the A condition to "jump-start" her performance in the B condition, similar to Experiment 1. Figure 3 shows the average learning curves we observe in this experiment. Our transfer ratio r is 1.531, with a p-value of 0.0034. Table 1 summarizes the data we observed from the three experiments and provides confidence intervals for the transfer ratios.

References

Chang, Y., Morrison, C. T., Kerr, W., Galstyan, A., Cohen, P. R., Beal, C., St. Amant, R., & Oates, T. (2006). The jean system. *Proceedings of the Fifth International Conference on Development and Learning (ICDL 2006).*

Cohen, P. R. (1995). *Empirical methods for artificial intelligence.* Cambridge, MA: The MIT Press.

Cohen, P. R. (1998). Maps for verbs. *Proceedings of the Information and Technology Systems Conference, Fifteenth IFIP World Computer Conference.*

Johnson, M. (1987). *The body in the mind: The bodily basis of meaning, imagination, and reason.* Chicago, IL: University of Chicago Press.

Lakoff, G. (1987). *Women, fire and dangerous things.* Chicago, IL: University of Chicago Press.

Mandler, J. (2004). *The foundations of mind: Origins of conceptual thought.* Oxford University Press.

Oakley, T. (2006). Image schema. In D. Geeraerts and H. Cuyckens (Eds.), *Handbook of cognitive linguistics.* Osford University Press.

Piaget, J. (1954). *The construction of reality in the child.* New York: Basic.

St. Amant, R., Morrison, C. T., Chang, Y., Cohen, P. R., & Beal, C. (2006). An image schema language. To appear in the Proceedings of The 7th International Conference on Cognitive Modelling (ICCM 2006).

Talmy, L. (2003). *Toward a cognitive semantics*, vol. 1: Conceptual Structuring Systems (Language, Speech and Communication). Cambridge, MA: The MIT Press.

Thelen, E., & Smith, L. (1994). *A dynamic systems approach to the development of cognition and action.* Cambridge, MA: The MIT Press.