Action Representation, Prediction and Concepts

Michael T. Rosenstein, Paul R. Cohen, Matthew D. Schmill, and Marc S. Atkin
Computer Science Department, LGRC
University of Massachusetts
Box 34610
Amherst, MA 01003-4610
{mtr, cohen, schmill, atkin}@cs.umass.edu

Abstract

A conceptual framework is a valuable resource for planning by situated agents. In this paper, we discuss the acquisition of such a framework. We take the position that concepts are abstractions of experience that confer a predictive ability for new situations. We also show specific examples which demonstrate the utility of abstract representations of actions, called *activity maps*, for reasoning about concepts and their entailments. In fact, we make the case that activity maps are concepts themselves. Where appropriate, we draw analogies with related work in nonlinear dynamics.

Introduction

Reasoning about one's environment, possible actions, and desired outcomes is the basis for planning — especially interactionist or improvisational planning (Agre & Chapman 1990). Although our current focus is not planning per se, we do address the prerequisite problem of acquiring a conceptual structure for planning. In particular, this research is part of an effort to explain how sensorimotor agents develop symbolic, conceptual thought, as every human child does.

Like Brooks (Brooks 1991) and others, we are trying to "grow" an intelligent agent from minimal beginnings by having it interact with a complex environment (the "Baby Project" (Cohen *et al.* 1996)). A problem for these projects is the transition from sensorimotor programs to symbolic concepts. It's one thing to store a motor scheme for shaking a rattle, quite another to have the symbolic concept RATTLE, with all its entailments (i.e., plausible inferences), or even an extensional category of rattles. Thelen and Smith (Thelen & Smith 1994), Kiss (Kiss 1991), Smithers (Smithers 1995), and Steels (Steels 1995) suggest that concepts (or categories, it isn't always clear which is intended) might be represented as dynamical systems, but none of these researchers demonstrates how such representations might be learned and used, or what their properties are.

Just as dynamicists illustrate system behavior through the use of delay portraits and Poincaré maps, we have developed "dynamical" representations of activities which we call *activity maps*. (We also make the distinction between two types of activity maps: *behavior maps* and *interaction maps*.) Our initial experimental domain is one in which two simple agents interact in a two-dimensional field. The movement of each agent is controlled by one of nine programs, and in most programs, movement depends on what the other agent is doing. For example, the AVOID program makes one agent move away from another, whereas the CRASH program makes an agent try to hit the other. When one agent runs the CRASH program and the other runs AVOID, the emergent behavior is often CHASE. It is relatively easy to recognize this combination of behaviors if one can see the entire interaction between the agents, but difficult to identify constituent behaviors from small snippets of the interaction. Surprisingly, activity maps can recognize what the agents are doing, quite accurately, with very little data.

Are activity maps useful for planning? Surely, if activity maps lead to a conceptual structure for reasoning about actions and outcomes. One hint to this possibility comes from Agre and Chapman's (Agre & Chapman 1990) *plan-as-communication* view of plans. In this view, a plan no longer plays a central role in specifying activity, but rather *guides* an agent in action selection. An integral part of plan-as-communication is a *theory of activity* which requires "two interconstraining parts: a theory of cognitive machinery and a theory of the *dynamics* or regularly occurring patterns of activity. (Agre & Chapman 1990)" Activity maps provide both of these things.

Are activity maps concepts? If by concept one means an abstraction that identifies a category and is invested with meaning through its predictions, then activity maps probably qualify. This issue is discussed further in a later section. Certainly, activity maps meet many requirements for *image schemas*, just as CONTAINER, PATH and ANIMATE-MOTION, which are thought by many researchers in philosophy, linguistics, and psychology to be prelinguistic chunks from which concepts are built and categories extracted (Lakoff 1984; Mandler 1992; Gibbs & Colston 1995). Roughly, the developmental argument goes like this: As we acquire image schemas, we use them to classify aspects of experience, learning schematic structures as a result. For instance, one may learn that rattles may be shaken by entities that display ANIMATE-MOTION, leading to a scheme in which the variable that we usually label "agent" must be filled by an entity capable of animate motion. To date, nobody has developed an AI program that learns concepts this way. The current work is a step in this direction.

The Simulator

A type of billiard ball simulator was created in which the balls may be viewed as reflexive agents endowed with one of nine possible behaviors:

1. NONE. Like conventional billiard balls, the agent's path is a line determined by its initial velocity. No attention is paid to the agent's "opponent."
2. AVOID. The agent attempts to move away from its opponent and avoid contact. Actually, we implemented three versions of this behavior: weak (AVOID-), normal (AVOID), and strong (AVOID+).
3. CRASH. The agent attempts to move toward its opponent and initiate contact. As with AVOID there are weak, normal, and strong versions (CRASH-, CRASH, and CRASH+, respectively).
4. KISS. Similar to CRASH except that the agent slows down before making contact. The appearance is that of a light touch.
5. RANDOM. The agent performs a smoothed random walk without regard to its opponent.

Figure 1 shows two examples from the simulator. For each trial, the playing field is an infinite plane and the graphical display pans to give an indication of the agents' relative positions. Two circles mark the boundaries for the *region of interest* (the outer circle) and the *region of interaction* (the inner circle). Within the region of interaction, the agents exhibit their predetermined behaviors; otherwise, they move with constant (randomly chosen) velocity, i.e., they behave as NONE. A trial terminates when 1) the trial duration exceeds a time limit which is large compared to the duration of a typical interaction, 2) the agents make contact, or 3) the agents exit the region of interest.

The simulator implements each behavior by varying the acceleration of one agent relative to the other. Acceleration, in turn, is given by a deterministic function of *proximity* (the distance between two agents) and *divergence* (the rate of change in proximity). Hence, one may interpret acceleration as the agent's *attempt* to change proximity. It follows that a behavior in the billiards domain is simply an equation of the form

$$attempt = f(proximity, divergence), \qquad (1)$$

where we define a positive *attempt* as an acceleration away from an opponent or, equivalently, an attempt to increase the gap between agents. (See Table 1 for details.)

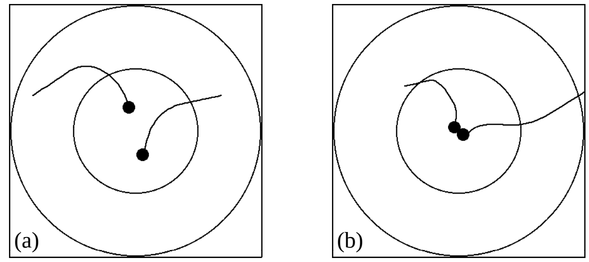

Figure 1: Simulator screen dump showing a representative trial of: (a) AVOID vs. CRASH; (b) KISS vs. KISS.

Behavior	*Implementation*
NONE	$a = 0$
AVOID	$a = 0.001/p^2 - 0.05 \cdot min(0, d)$
AVOID-	$a = 0.5 \cdot \text{AVOID}$
AVOID+	$a = 2.0 \cdot \text{AVOID}$
CRASH	$a = -0.0075 \cdot p^2 - 0.05 \cdot max(0, d)$
CRASH-	$a = 0.5 \cdot \text{CRASH}$
CRASH+	$a = 2.0 \cdot \text{CRASH}$
KISS	$a = \begin{cases} \text{CRASH} & \text{if } p > 0.5, \\ \text{AVOID} & \text{if } p < 0.5 \text{ and } d < 0.12, \\ 0 & \text{otherwise.} \end{cases}$
RANDOM	$a = \text{Uniform}(-0.01, +0.01)$

Table 1: Simulated behaviors and their implementation, where p is proximity, d is divergence, and a is attempt. Distances are normalized by the diameter of the region of interaction, and one unit of time corresponds to one iteration of the simulation equations.

Behavior Representation

The functions in Table 1 are compact representations of agents' behaviors, but in general such functions are unknown and must be estimated from observations of an agent's interaction with its environment. In fact, it is considerably easier to describe behavior by plotting *attempt* with respect to *proximity* and *divergence* than it is to estimate f in (1). Essentially, this is the purpose of attractor reconstruction methods which take time series and produce a topologically equivalent spatial representation of the underlying dynamical system. (See (Rosenstein, Collins, & De Luca 1994) and references therein.) The remainder of this section describes the construction of such representations, which we call *behavior maps*.

One could certainly devise a number of visualization techniques (e.g., vector fields, 3D surface plots) for creating activity maps—behavior maps in particular. For the present work, a decision was made to use grayscale images in which dark regions of the image indicate weak attempts by the agent and bright regions indicate strong attempts. Positive and negative signs distinguish attempts in favor of divergence from those aimed at shrinking the distance between the agents.

(Unvisited territory is shown in white with no sign.)

Figures 2 and 3 illustrate the behavior maps for AVOID and CRASH. These maps confirm our intuition about each behavior: For AVOID the attempt to increase proximity is greatest when the agents are near, although we see an abatement as the divergence changes from *closing* to *separating.* Conversely, CRASH is most aggressive when the agents are distant and separating.

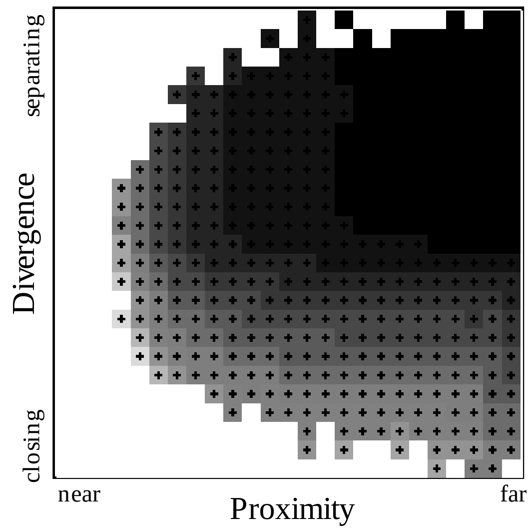

Figure 2: Behavior map for AVOID.

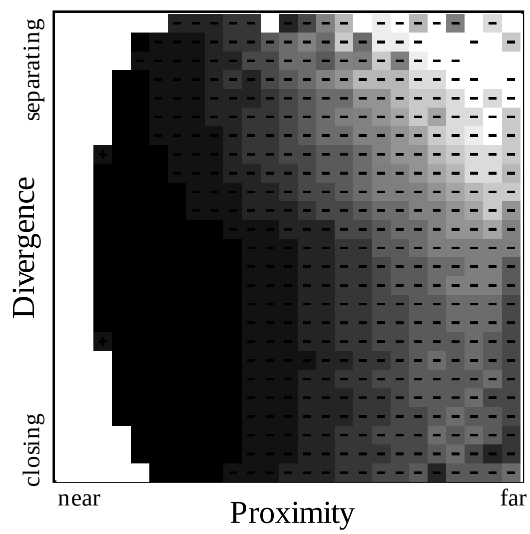

Figure 3: Behavior map for CRASH.

Behavior Recognition

To bridge the gap between behavior representation and behavior recognition, we need a technique for comparing pairs of maps. We compute a *difference map* by subtracting the values in corresponding cells of two maps. The resulting image depicts regions of similarity with dark gray values and differing regions with light values. For instance, the difference maps for KISS & CRASH and for KISS & AVOID suggest that KISS and CRASH are more alike than KISS and AVOID. Moreover, one easily identifies the portions of each map where there is the most, or least, similarity. To quantify these effects, we compute the *mean gray value*, $\overline{gv}$, for the difference map:

$$\overline{gv}(map1, map2) = \frac{1}{N}\sum_{i}^{N} |gv_i^{(map1)} - gv_i^{(map2)}|, \tag{2}$$

where $gv_i^{(map1)}$ and $gv_i^{(map2)}$ are corresponding gray values from the two original maps, and i indexes into the desired cells. Note that for the aforementioned difference maps, the normalized values for $\overline{gv}$ are 0.2140 (KISS - AVOID) and 0.0851 (KISS - CRASH), with extreme values of 0 and 1 indicating identical and completely dissimilar maps, respectively.

For behavior recognition, the task is to compare not one behavior map with another, but rather an agent's behavior *trajectory* with the maps from known behaviors. Here, a behavior trajectory refers to a sparse behavior map constructed from a limited period of observation. One may still use Eq. (2) with i indexing over the cells of the trajectory.

Just as humans are thought to acquire image schemas over time, the billiards simulator accumulates a library of behavior maps, one for each of the nine agent behaviors. Each constituent map is built up through experience by having one agent type interact repeatedly with the behavior program for RANDOM. Then, for every post-training trial, a *recognizer* measures the time-varying positions of the agents, computes a pair of trajectories, and evaluates Eq. (2) for each trajectory and every image in the behavior library. For each trajectory, the recognizer's response is simply the behavior that results in the smallest value for $\overline{gv}$. Table 2 shows the responses — along with the true behaviors — from numerous trials where the agent behaviors were selected randomly.

Table 2 is a *confusion table* that demonstrates the recognizer's misinterpretations between all eighty-one possible pairs of behaviors. For example, when the agent is of type AVOID, the recognizer makes the correct response for 72% of the trials and chooses one of the other forms of AVOID almost all other times. Note that these data were derived in the presence of simulated measurement noise; not shown is the noise-free case where we see correct responses (values along the diagonal) much closer to 100%. Interestingly, the recognizer performs better when the actual behavior is any flavor of AVOID than it does when the true behavior is one of the CRASH variants. Our explanation is this: For an interaction to occur in the first place, two agents must approach one another, i.e., they must have negative divergence. Moreover, when one agent exhibits a CRASH behavior, the initial gap closes at an even greater rate — rapidly placing the CRASH trajectory into the left half of the behavior map where proximity is small. For all behaviors except the AVOID programs, a small value for proximity corresponds to a weak attempt. Hence, these behaviors look very much alike in the left half-plane. In contrast, the agents' initial convergence facilitates the identification of AVOID programs by placing the trajectory in the most discriminating region of those maps. Though not shown by these data, we suspect that the least confusion among CRASH behaviors occurs when the interaction is with one of the AVOID agents; in these instances, the trajectory remains in the right half-plane for a greater portion of the observation time.

	Recognizer Response								
Actual	N	A-	A	A+	C-	C	C+	K	R
N	**31**	16	5	2	20	2	0	0	23
A-	17	**50**	18	5	5	0	0	0	5
A	2	15	**72**	8	0	0	0	0	3
A+	0	2	8	**90**	0	0	0	0	1
C-	18	4	2	1	**24**	16	0	4	30
C	2	2	0	1	18	**21**	18	23	14
C+	0	0	0	0	1	9	**76**	13	1
K	7	1	0	0	18	17	22	**22**	12
R	12	13	9	11	10	11	3	13	**15**
All	10	11	12	14	11	9	14	9	12

Table 2: Confusion table for 500 interactions with agent behaviors chosen randomly. Shown are response percentages for each behavior given an observation time of 5 simulation cycles. Position measurement noise was simulated by superposition of uniformly distributed noise with a mean of zero and a range equal to 1% of the diameter of the region of interaction. Behavior names are shortened to first letters only, and - and + indicate weak and strong forms, respectively.

Interactions and Category Prediction

To make the case that activity maps are concepts, as we do in the next section, we must show that activity maps provide the "infrastructure" for predictive inferences. Specifically, behavior maps help us recognize a particular set of circumstances, and this sets the stage for reasoning about categories and outcomes. To accomplish the latter task, we make use of *basins of attraction.*

Figure 4 is a schematic of the basins of attraction for interactions between AVOID and CRASH. (We call this diagram the AVOID/CRASH *interaction map.*) A basin of attraction is the set of all initial conditions which lead to a particular *limit set*, i.e., post-transient system behavior. The billiards system has three stable limit sets where net attempt—the sum of the attempts of the two agents—eventually becomes zero. We label these sets, respectively, CONTACT, CHASE, and ESCAPE. Also shown in Figure 4 are representative *interaction* trajectories leading to each limit set. (Readers concerned by the trajectory crossings should note that these particular maps are projections of eight dimensions onto two and uniqueness is not violated in the higher-dimensional map.)

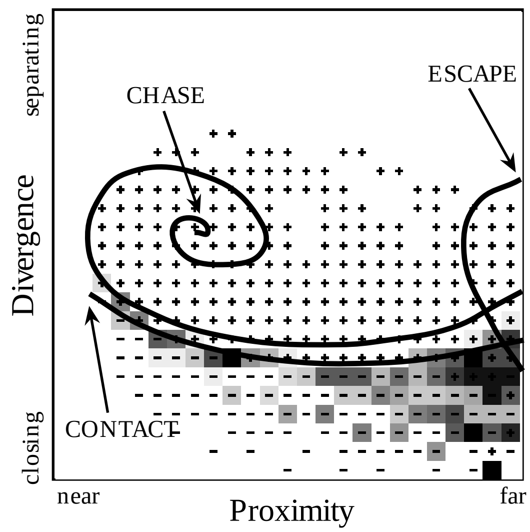

Figure 4: Interaction map for AVOID interacting with CRASH. Positive and negative signs indicate, respectively, NO-CONTACT and CONTACT as the final outcomes, and gray cells connote uncertainty.

Interaction maps are learned in a similar fashion as for behavior maps. In numerous trials, one agent type interacts with another (AVOID versus CRASH in this example), a trajectory is computed for each interaction, and the cells in the trajectory are colored not by attempt, but by final outcome. For simplicity, CHASE and ESCAPE are combined in a limit set called NO-CONTACT. Every cell in a trajectory that deterministically ends in a no-contact situation is labeled "+". Conversely, "-" denotes the limit set where agents deterministically make contact. Gray values represent probabilistic outcomes: lighter grays represent higher probabilities for the outcome denoted by the sign in the cell.

Interaction maps allow agents to reason about two *entailments* of each scenario, namely, whether or not they will make contact. We built an *interactions library* with one entry for each possible pair of behaviors. Reasoning then becomes a two-step process: first, *recognize* the agents' behaviors and select the corresponding entry in the interactions library; second, *predict* contact or no contact based on the information in the interaction map. As they provide two opportunities for error, we discuss these steps in turn.

Figure 5 shows the recognizer's performance for interactions between AVOID+ and CRASH+. When the observation time is short, the recognizer confuses the different crash programs roughly one-third of the time. Notice, however, that recognition performance improves as the situation unfolds; this suggests another dimension along which an agent might reason: *act now* or *wait* for more definitive information which may or may not appear? The data in Figure 5 seem to make the case for waiting; recognition is nearly perfect after 20 simulation cycles. However, these data only tell half the story. Figure 6 shows that the predictive power gleaned from the interaction maps also varies as a function of observation time. Accuracy actually drops and then recovers. In a future experiment, we will combine these two pieces of information, with the goal of creating an agent that can reason about its environment and modify its behavior accordingly. For example, we would like an AVOID agent to predict a future CONTACT situation and take the necessary action (at a cost to the agent) to "jump" itself out of an undesirable basin of attraction.

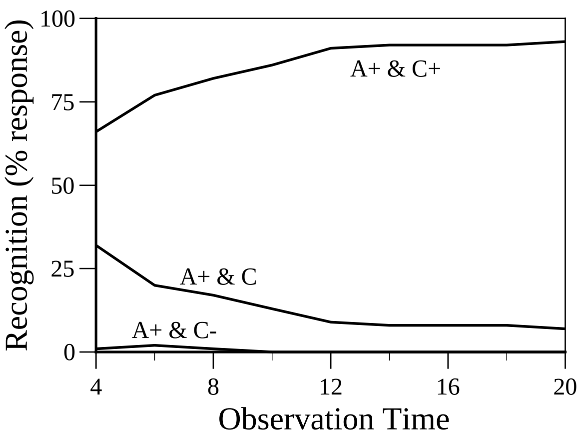

Figure 5: Recognizer responses for AVOID+ vs. CRASH+. Each curve gives the frequency that the recognizer responded with a particular pair of behaviors.

Concepts and Categories

Here we argue the case that activity maps are concepts. At first this equivalence is hard to accept, especially if one thinks of concepts as collections of necessary and sufficient, objective features. On the other hand, if one thinks of concepts as abstractions of regularities in experiences which may be used for classification and prediction, then activity maps qualify. Let us define experiences to be trajectories through a space of very high dimension; for example, the experience of positioning oneself and sitting in a chair involves visual, somatic,

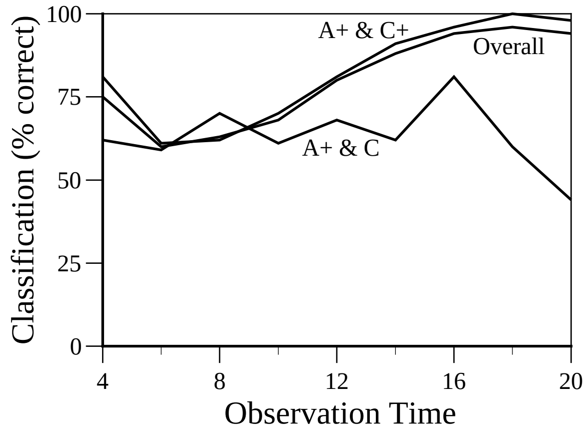

Figure 6: Classification performance for AVOID+ vs. CRASH+. The "overall" curve represents the percentage of correct predictions irrespective of whether recognition was correct.

kinesthetic and affective dimensions. Perhaps we store all these streams—perhaps we maintain internal representations of the fine details of specific experiences—but for the purposes of differentiating and predicting experiences, this isn't necessary. One can imagine an internal representation of sitting in chairs sufficient to differentiate it from, say, sitting in saddles. The predictions associated with these representations are different, too: Chairs tend to hold still whereas saddles (on horses at least) bounce around, and different parts of one's anatomy are apt to get tired. Similarly, one can imagine an internal representation of perching on dining chairs that is sufficient to differentiate it from, say, relaxing in arm chairs; and these representations also would predict different things about the activities.

Which abstractions should an agent acquire? The answer has two components: First, which distinctions and predictions are important to the agent, and second, which abstractions of experience provide a good basis for making these distinctions and predictions? In polite society, one is rewarded for observing a distinction between dining chairs and easy chairs—for maintaining a Victorian posture in the former—but not for finer distinctions within either class. It suffices to have abstractions "how I sit in a dining chair" and "how I relax in an easy chair." These abstractions must contain dimensions that differentiate activities; for instance, postural information. And equally significant from the standpoint of rewards, the abstractions should contain dimensions that help the agent predict outcomes. Importantly, agents in different environments with different reward structures won't acquire the same abstractions. They maximize reward if abstractions are good bases for distinctions and predictions, but abstractions themselves depend on the agents' experiences. This is just another way of saying we favor an interactionist rather than objectivist epistemology (Lakoff 1984;

Johnson 1987).

Abstractions of experience immediately identify *categories* of experience and, thus, categories of entities that play roles in experiences. Indeed, objects and experiences seem to be duals: A category of experiences can identify a category of objects, or vice versa. The concept "chair" identifies sitting experiences, and if one wishes to differentiate these experiences then one must often differentiate the chairs.

Now we would like to substitute the word "concept" for "abstraction." Concepts, then, are abstract representations of experience that are acquired (over other concepts) for their ability to make important discriminations and predictions, and which identify categories of experiences and objects. Some would protest that this definition doesn't say what a concept *means*, but we would disagree. We propose a *predictive semantics* where the meaning of a concept is the predictions it makes. In fact, predictiveness underlies our entire approach to concept acquisition: Concepts are abstractions selected for their ability to differentiate and predict, and their meanings are just their predictions. This is one interpretation of Mandler's claim (Mandler 1992) that concepts are "minitheories" (i.e., predictive statements) and Lakoff's (Lakoff 1984) and Johnson's (Johnson 1987) observations that image schemas (abstractions of experience) support *entailments* (i.e., predictive inferences).

On this account, activity maps are concepts and their meanings are the predictions they support. To complete the analogy with nonlinear dynamics, concepts and entailments are like the topological invariants we wish to preserve when we decide upon a visual representation for activity. A behavior map is an abstraction in two senses: first, it represents experiences in a projection of just two of many dimensions (*proximity* and *divergence*); second, a map selects a class of experiences that is larger than the union of the experiences recorded in the map. It is possible to plot a trajectory through a map that has never been observed. It is also possible to observe a new trajectory and assess its degree of match to a given map (with $\overline{gv}$). The meaning of a map, say CRASH, is the predictions it makes about whether or not the balls will collide. Necessarily, meaning will depend on the context provided by an experience. For instance, if the experience takes the agent into a basin of attraction in which collision is inevitable but hasn't yet occurred, then the meaning of the associated concept, CRASH, is that a collision is inevitable. Just as the concept CHAIR means different things in different contexts, so does the meaning of CRASH.

Let us comment on some other aspects of concepts and categories and how they correspond to aspects of activity maps. Some instances of concepts are judged by humans to be "better examples" than others; Rosch discovered that robins, for example, are good prototypes for birds whereas turkeys are not (Rosch & Lloyd 1978). A corresponding notion is that some trajectories through activity maps occur more often than others. Perhaps such a trajectory would be judged a prototypical CRASH, say. Rosch also discovered that humans organize categories into *basic, superordinate* and *subordinate* levels (e.g., dog is basic, pet is superordinate, spaniel is subordinate). We think that the underlying theme of predictiveness can explain the distinction: Basic level categories contain objects that are specified in enough detail to predict experience, thus because dogs differ from cats in their behavior, "pet" is not a sufficiently predictive designation and "spaniel" doesn't provide enough additional predictive power (for most people in most contexts) to be a distinction worth making.

One aspect of concepts is yet to be accounted for by activity maps. Concepts are related to others in thematic structures, and one can pick out the roles in these structures. For example, two youths fall in love, their parents object, both youths die, and we call this a classical tragedy. When we see it written down, we can point to the actors and their relationships; tragedies have rich internal structure. Activity maps are abstractions of experience, and while we might be able to identify roles (e.g., the ball that crashes, the one that avoids), it is not yet clear whether we can compose the experiences captured by a map with other experiences and still be able to point to the components. That is, we have yet to learn whether activity maps can have rich internal structure that we can point to and reason about. Nevertheless, this work has taken us a step closer to an understanding of "concept." Indeed, we are encouraged by how far we've come with activity maps from such a simple experimental domain.

Planning

Under Agre and Chapman's (Agre & Chapman 1990) *plan-as-communication* paradigm, one views an agent as a flexible participant in the world. Planning occurs during the course of action and requires the repeated evaluation of the current situation with respect to future outcomes. Behavior maps and interaction maps provide a means for this very sort of participatory planning.

As an example, consider an interaction between AVOID- and CRASH. After behavior recognition, suppose the AVOID- agent finds itself in the state labeled **A** in the interaction map of Figure 7a. Without a change in circumstances this state assures a CONTACT outcome—something AVOID agents dislike. As a response, AVOID- has two qualitatively different options: state perturbation or parameter adjustment. (These possibilities are labeled states **B** and **A'** in Figures 7a and 7b, respectively.) State perturbation refers to a short-lived effort by AVOID- to alter its location in the interaction map. For example, a rabbit may slow its predator temporarily by darting through a small opening in the brambles. Parameter adjustment, on

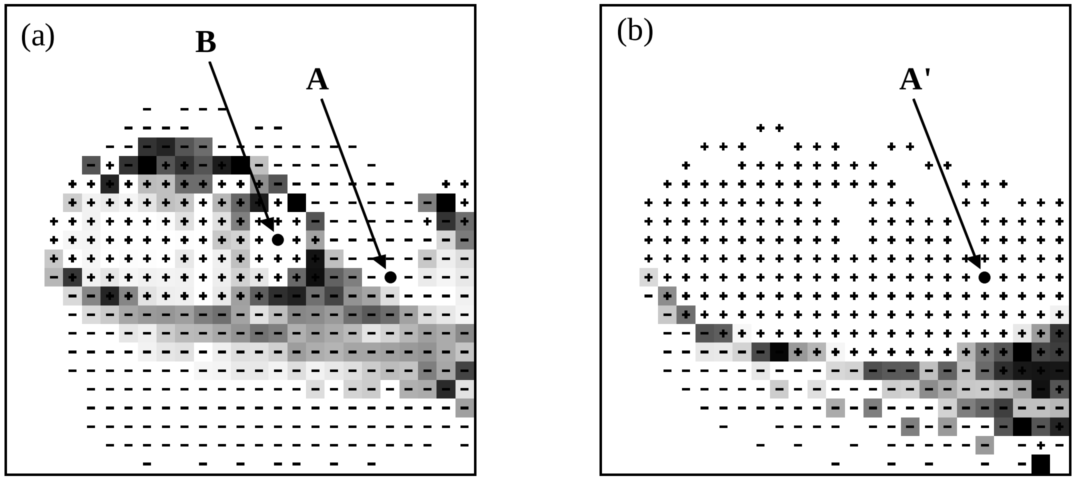

Figure 7: Interaction maps for CRASH interacting with (a) AVOID- and (b) AVOID. Positive and negative signs indicate, respectively, NO-CONTACT and CONTACT as the final outcomes. Given current state **A**, the AVOID agent may attain NO-CONTACT by qualitatively different actions leading to state **B** or to state **A'**.

the other hand, is akin to a surge in adrenaline which transforms our bunny into virtual jackrabbit, i.e., a more intense AVOID agent which brings a different interaction map into play.

In the previous example, we say nothing about how an agent actually *decides* what actions to follow. This problem is beyond the goals of this paper, although we are about to build agents that select actions by visualization (Agre & Chapman 1990), i.e., by imagining the states and trajectories produced by available actions. Behavior maps and interaction maps naturally lend themselves to visual manipulation (e.g., $\overline{gv}$, cluster analysis) and are ideal representations for planning by visualization.

Acknowledgements

We thank T. Oates and D. Westbrook for stimulating discussions. This research is supported by DARPA under contract No. N66001-96-C-8504. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright notation hereon. The views and conclusions contained herein are those of the authors and should not be interpreted as the official policies, either expressed or implied, of DARPA or the U.S. Government.

References

Agre, P. E., and Chapman, D. 1990. What are plans for? *Robotics and Autonomous Systems* 6:17–34.

Brooks, R. A. 1991. Intelligence without Reasoning. In *Proceedings of the Twelth International Joint Conference on Artificial Intelligence*, 569–595.

Cohen, P. R.; Oates, T.; Atkin, M. S.; and Beal, C. R. 1996. Building a baby. In *Eighteenth Annual Conference of the Cognitive Science Society*, 518–522.

Gibbs, G. J., and Colston, H. L. 1995. The cognitive psychological reality of image schemas and their transforms. *Cognitive Linguistics* 6–4:347–378.

Johnson, M. 1987. *The Body in the Mind.* University of Chicago Press.

Kiss, G. 1991. Autonomous agents, AI and chaos theory. In an S.W. Wilson, J.-A. M., ed., *From Animals To Animats: Proceedings of the First International Conference On the Simulation of Adaptive Behavior*, 518–524. Cambridge Massachusetts: The MIT Press.

Lakoff, G. 1984. *Women, Fire, and Dangerous Things.* University of Chicago Press.

Mandler, J. M. 1992. How to build a baby: II. Conceptual primitives. *Psychological Review* 99(4):587–604.

Rosch, E., and Lloyd, B. B. 1978. *Cognition and Categorization.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Rosenstein, M. T.; Collins, J. J.; and De Luca, C. J. 1994. Reconstruction expansion as a geometry-based framework for choosing proper delay times. *Physica D* 73:82–98.

Smithers, T. 1995. Are autonomous agents information processing systems? In Steels, L., and Brooks, R., eds., *The Artificial Life Route To Artificial Intelligence: Building Embodied, Situated Agents.* Lawrence Erlbaum Associates. 123–162.

Steels, L. 1995. Intelligence - dynamics and representations. In Steels, L., ed., *The Biology and Technology of Intelligent Autonomous Agents.* Berlin: Springer Verlag.

Thelen, E., and Smith, L. 1994. *A Dynamic Systems Approach to the Development of Cognition and Action.* Cambridge, MA: MIT Press.