# *Object schemas for responsive robotic language use*

**Citation:** Hsiao, Kai-yuh, Soroush Vosoughi, Stefanie Tellex, Rony Kubat, and Deb Roy. “Object Schemas for Responsive Robotic Language Use.” Proceedings of the 3rd International Conference on Human Robot Interaction - HRI ’08 (2008), March 12-15, 2008, Amsterdam, the Netherlands, ACM Press, (2008).

**As Published:** http://dx.doi.org/10.1145/1349822.1349853

**Publisher:** Association for Computing Machinery

**Persistent URL:** http://hdl.handle.net/1721.1/86878

**Version:** Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

# Object Schemas for Responsive Robotic Language Use

Kai-yuh Hsiao
MIT Media Lab
Cambridge, MA, USA
eepness@media.mit.edu

Soroush Vosoughi
MIT Media Lab
Cambridge, MA, USA
soroush@media.mit.edu

Stefanie Tellex
MIT Media Lab
Cambridge, MA, USA
stefie10@media.mit.edu

Rony Kubat
MIT Media Lab
Cambridge, MA, USA
kubat@media.mit.edu

Deb Roy
MIT Media Lab
Cambridge, MA, USA
dkroy@media.mit.edu

## ABSTRACT

The use of natural language should be added to a robot system without sacrificing responsiveness to the environment. In this paper, we present a robot that manipulates objects on a tabletop in response to verbal interaction. Reactivity is maintained by using concurrent *interaction processes*, such as visual trackers and collision detection processes. The interaction processes and their associated data are organized into *object schemas*, each representing a physical object in the environment, based on the target of each process. The object schemas then serve as discrete structures of coordination between reactivity, planning, and language use, permitting rapid integration of information from multiple sources.

## Categories and Subject Descriptors

I.2.9 [**Artificial Intelligence**]: Robotics; I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*Language parsing and understanding*

## General Terms

algorithms, design

## Keywords

robot, object schema, behavior-based, language grounding, affordances

## 1. USING LANGUAGE RESPONSIVELY

Speech interaction allows humans to convey complex beliefs and desires efficiently. Thus, speech interfaces are a natural candidate for making robots useful to the general public in a flexible and autonomous way. However, language use is an inherently discrete, symbolic task, while robots exist in a continuous world of noisy sensorimotor data. The problem of *language grounding*, connecting words to the real world, has been addressed by a number of researchers (see [27, 28], and Section 4 below). Most implemented approaches extract discrete models and labels from sensor data as a foundation for symbolic language use.

However, models and labels thus constructed run the risk of falling out-of-sync with the immediate sensorimotor environment. A dynamic physical environment might change while the robot is performing verbal or physical interactions according to its internal model, leading to poor responsiveness. Our goal is to address language grounding in a robot, such that the discrete structures used for language and planning stay as tightly bound as possible to the continuous sensorimotor level, accomplishing the responsiveness typically found in *behavior-based* robots (e.g., [7, 9], or other examples given in Section 4). We accomplish this by using an *object schema* representation to coordinate between discrete aspects such as language and planning and the continuous aspects of responding to a dynamic sensorimotor environment.

### 1.1 Overview of the System

Our robot system is designed to handle natural language requests while being responsive to, and even leveraging, changes to its knowledge of the physical environment. The central point of coordination is our object schema representation, each of which represents information about one physical object in the robot's environment. Each object schema consists of a bundle of multiple *interaction processes* (or just "processes," in this context), which are concurrently-running execution loops that coordinate either sensorimotor activity such as visual tracking, or internal state management such as devising a plan for action. The processes bundled in an object schema are the processes that are, or could be, acting upon the represented object. For instance, for a given physical object, the processes that visually track it, attempt to grasp it, and coordinate plans to move it are all linked together. Because the interaction processes constitute actions that are or could be acting on an object, each object schema can also be viewed as a representation of *affordances*, the actions and results enabled by each object [19].

The object schemas act as discrete entities for the purposes of language and planning. Organizing continuous processes into object schemas allows incoming sensory data to be readily sorted for rapid interaction with language and planning, and vice versa. It should be noted that the term "schema" is used here in a psychological sense, as used by Piaget [25], or

*HRI'08,* March 12–15, 2008, Amsterdam, The Netherlands.
Copyright 2008 ACM 978-1-60558-017-3/08/03 ...$5.00.

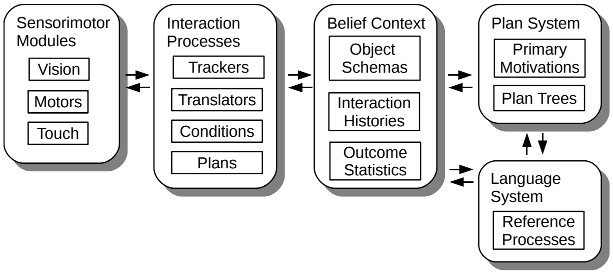

**Figure 1: Simplified block diagram of the system, showing data flow between the five parts of the system. Note that most of the data flow occurs by virtue of object schemas and plan trees being composed of interaction processes, which is not depicted.**

computationally by Drescher [14] or Roy [28], in that incoming continuous sensory percepts are organized into discrete structures. In this case, incoming sensory data is regarded as being signs of objects in the environment. Our object schemas are an implementation based on our interpretation of the term in this context.

Our robot, named Trisk (Figure 2 shows the robot in action), is a six-DOF (degree of freedom) robotic arm with a four-DOF Barrett Hand as its end effector, situated in front of a table on which manipulable objects are placed. Six-axis force-torque sensors on each of the three fingers of the hand enable sensing of forces and collisions, including awareness of successful grasps. Two cameras (only one is currently active for simplicity) sit in a head mounted on a four-DOF neck, which allows the system to adjust its view of the environment and look up at the human for interactivity purposes.

Some of the behaviors of the system include:

**Responsive behaviors** The system is capable of responding rapidly to certain sensory triggers, by moving away from sensed collisions and also using knowledge of collisions to adjust grasping targets. The object schemas serve to coordinate visual, touch, and motor-related data in such cases.

**Verbal behaviors** The system also handles straightforward verbal requests, such as "Touch the red ball," or "Move the block to the right." The object schemas provide a connection between discrete verbal referents and the resulting motor actions in the continuous sensory world.

**Responsive verbal behaviors** The system can responsively carry out verbal requests such as "Move the block to the right... No, to the left." The interaction with this verbal input is "responsive" because it requires immediate interruption and revision of the current plan due to new input activity.

All of the interactions are the result of the interplay between five key parts of the system (also depicted in Figure 1), with the object schemas serving as the main point of interconnection:

1. The robot platform and its sensorimotor modules, which handle vision, touch, speech, and motor control.

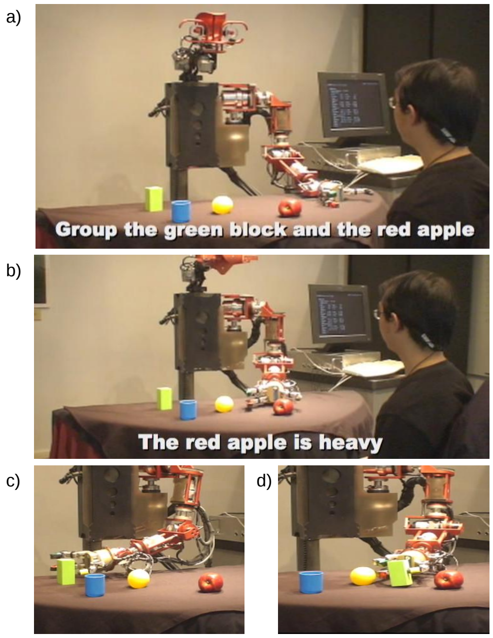

**Figure 2: The robot is facing a scene that includes a red apple and a green block. a) The robot is told, "Group the green block and the red apple." This request could be satisfied by moving the block towards the apple, or vice versa. The robot decides to move the apple. b) While the robot reaches for the apple, the human adds, "The red apple is heavy." Knowing that heavy objects are more difficult to lift, the robot changes its plan and c) moves the green block d) towards the apple instead.**

2. The interaction processes, which run concurrently to perform tasks like visual tracking and action monitoring.

3. The *belief context*, which organizes interaction processes and their related data into object schemas, each representing one object, its attributes (such as color, shape, and weight), and its affordances (such as liftability and graspability).

4. The *planning system*, which reads information from the belief context, devises plans, and coordinates their execution and revision.

5. The *language system*, which converts linguistic inputs into plans and object-centered knowledge.

## 1.2 Sample Interaction

Figure 2 shows an example of an interaction with the robot. In this section we briefly explain the mechanism underlying the example given in the figure. The robot is facing a scene in which a red apple and a green block, among other objects, are on the table. The visual inputs coming from the cameras are sent to a color-based segmenter, which extracts regions of uniform color. Each of these regions leads to the creation of an interaction process that monitors subsequent visual frames for similar regions, i.e., the process "tracks" the object from frame to frame.

These *tracking processes* are incorporated in separate object schemas, denoting that they are targeting different objects. A series of *translation processes* are then created, which acquire and categorize shape, size, and location information about the objects. One such translation process converts the locations in 2-D camera space into locations in 3-D arm space, based on the assumption that objects are at the hard-coded table height. The translated locations serve as target locations for physically reaching towards the objects.

Next, the human says "Group the green block and the red apple." The verbal input is processed by a speech recognizer and a parser, which outputs a structured parse tree for translation into the system. The two noun phrases, "green block" and "red apple," give rise to *reference processes*, which are interaction processes that search the current set of object schemas for matches based on category labels. The verb, "group," gives rise to a plan tree in the planning system that takes the matches from the reference processes and searches for a sequence of actions that will lead to the targeted objects being grouped together.

There are two ways to satisfy "Group the green block and the red apple": either the robot can lift the red apple and place it near the green block, or it can lift the green block and place it near the red apple. The planning system selects between these alternatives based on prior data about objects with the given attributes (green, red, block, and apple). The plan tree is constructed using the choice with the highest predicted success likelihood. The planning system's search will produce a series of *action processes*, which are interaction processes that issue motor commands and monitor their progress.

In this example, the robot opts to reach for the red apple. At this point, the human says, "The red apple is heavy." The verbal input is processed such that the object schema for the red apple takes on the additional attribute *heavy*. This attribute is known to the planner (based on its data about attributes) to result in a poor chance of success for manipulation. The planner immediately revises its plan based on the new information, and the robot lifts its hand and reaches towards the green block instead. The robot lifts the green block and places it next to the red apple.

Now, suppose the robot had failed to grasp and lift the green block four times in a row. At this point, the collected statistics for the liftability of the green block, accumulated as part of its object schema, would outweigh the data about heavy objects, and the planner would once again revise its plan. The robot would then reach for the red apple again instead, grasp it, and move it next to the green block.

# 2. IMPLEMENTATION

In this section we describe the parts of the system that interact via the object schemas: the sensorimotor modules, the interaction processes, the belief context, the planning system, and the language system.

## 2.1 Sensorimotor Modules

Visual input from the active camera is sent through a color-based segmentation algorithm (based on CMVision [8]) that groups contiguous regions by color. The current set of objects used for robot interactions consists of simple objects of uniform color, so color segmentation suffices for our purposes. Visual input is also processed on demand by a mean-shift tracker [13] based on edge and color profile, and a 2-D shape recognition algorithm based on shape contexts [3], when requested by vision-related interaction processes. Visual information thus derived includes the size, shape, color, and location of each object, all of which can be matched with verbal object descriptions.

The motor control modules for the robot's arm and hand compute forward and inverse kinematics, so the hand can be brought via a smooth trajectory towards reachable 3-D coordinates. The fingers can be spread to enable grasping, or moved together to tap an object. Touch input from the fingers is used along with arm kinematic information to provide the location, direction, and magnitude of contact forces between the fingers and physical objects.

Speech input is collected by a microphone headset worn by the human, and passed through the Sphinx 4 free speech recognizer before being processed by downstream modules in our system.

Vision, touch, speech, and motor control provide a rich foundation for responsive verbal interaction. The rest of the system is responsible for the necessary integration tasks to make coherent behavior possible.

## 2.2 Interaction Processes

At the core of the system's sensorimotor coordination are concurrently-running interaction processes. An interaction process writes data derived from its execution to an *interaction history*, which is kept in shared memory to be read by related processes. During each cycle, an interaction process reads from various interaction histories, performs some processing, and writes to its own interaction history.

Most interaction processes are targeted towards one physical object. Data in an interaction history is thus generally about one object, and both the interaction process and its history are thus part of the object schema that represents that physical object. An object schema consists of all the processes, history data, and expectations associated with a single object.

The specific processing performed by an interaction process depends on the *process class* from which it is *instantiated*. Upon instantiation, the interaction process is passed some parameters indicating the object schemas, if any, that it is to target. The top-level process classes:

**Sensory processes** monitor incoming sensory data and write relevant data to their interaction histories. For instance:

- A visual tracking process is assigned to a visual region, checks subsequent visual frames for a region with similar properties, and writes new region data to its history.
- A grasp tracking process writes location data for an object when the robot is believed to be grasping the object.
- A collision detection process determines when force sensors encounter large forces, and writes collision data to its history to trigger an action that moves away from the collision.

**Action processes** read various interaction histories and, when active, send motor commands to the robot. Examples include a process to move the robot away from a collision, or to grasp the fingers around an object.

**Condition processes** repeatedly assess whether a condition is true or not, and when not true trigger the planning system to search for plan fragments that can render them true.

**Plan fragment processes** operate in the planning system to coordinate a sequence of actions and conditions. For instance, the plan fragment for grasping an object requires conditions that the hand be open and moved to the object's location. Then, it triggers the action for closing the fingers.

**Translation processes** convert interaction history data to another form, such as a conversion from visual 2-D data to 3-D coordinates for the arm to target, or from continuous color information to a discrete color category.

**Coordination processes** perform process-level coordination by instantiating and detaching other processes to maintain the coherence of the object schema. For instance, if visual and touch tracking disagree about an object's location, the coordination process may detach one of the processes from the object schema. A detached process will find a new target, either by matching a pre-existing object schema or by triggering the creation of a new, empty object schema.

**Reference processes** receive noun phrases from speech input and attempt to connect the noun phrases to object schemas with matching attributes. For instance, "the green block" leads to a reference process that reads interaction histories of each object schema to find one that best fits the description.

After an interaction process is instantiated, it is ready to become *active* and run its associated code. However, some interaction processes are resource-bound, such as when multiple interaction processes want to control the arm or neck motors. Resource-bound interaction processes are activated by the planning system, based on their hand-coded priority. For instance, an action tracking process for moving away from a collision has a higher priority than any action tracking process that is addressing a verbal request.

## 2.3 The Belief Context and Object Schemas

The belief context consists of interaction processes, their interaction histories, and associated expectation data (discussed further at the end of this section), organized into object schemas. Each object schema's contents represent a set of beliefs that is probably about a real physical object. For instance, a visual tracker and its history data manages and contains a set of beliefs that include the location, color, and shape of an object. Similarly, an inactive action process for performing a grasp is associated with an expected success likelihood, which constitutes a belief about the graspability of the object.

The beliefs in an object schema are "probably about" a real object because sensor noise can lead to spurious object schemas, which sometimes is accounted for after repeated failures to touch and grasp the spurious object, but in the current implementation frequently leads to planning and language failures (discussed further in Section 3.2).

### 2.3.1 *Physical Objects from the Robot's Perspective*

In line with an affordance-based [19, 20] or schema-based view of objects [14], a physical object can be viewed (from the perspective of the robot) as nothing more than the processes that target it and the expected results of those processes. For example:

- As a visual tracking process checks subsequent frames for its target region, there is an implicit expectation that its target region will be found in similar form near its previous location in each new frame. Such a process thus sees only a target region and an expectation, and needs no concept of an "object."

- The visual location recorded by the visual tracking process is translated into a location in the arm's coordinate space. An action process targeting this location can expect to encounter a certain set of forces when moving near the location (i.e., when the hand bumps the object upon moving there). Once again, rather than referring to an "object," the interaction process only monitors and tracks a set of sensorimotor expectations.

Even though it may be convenient to speak of an object as the target of a interaction process, from the perspective of the interaction processes themselves, there exist only these expected interactions, which happen to relate to each other in such a predictable way that it is convenient to group them together and declare them an "object."

The notion of an object is thus initially constructed from the processes, results, and desires of the system. From this viewpoint, an object exists only in that a group of processes have a related set of expected results. On the other hand, having organized the processes into object schemas, it then becomes possible to manipulate the object schemas as discrete entities for the purpose of planning and communication. On one level of analysis, there are only interaction processes and expected results, but on another level of analysis, it is important to be able to point to a single discrete object. This premise sits at the core of our integration between the continuous sensorimotor world and discrete language and planning.

This process-centered representation provides a convenient means of organizing sensorimotor data and processes such that changes in one active process can immediately affect another process in that object schema. For instance, a moving visual region can rapidly change the target location for a grasp-related process. Likewise, a failed grasp can lead to doubt that a physical object actually exists at that location, leading in turn to reevaluation of the visual regions. This representation also provides a single point of connection for affordance-based terms, such as "liftable" or "graspable."

### 2.3.2 *Outcome Statistics*

Attributes of objects, such as color, shape, size, position, and weight, are written by interaction processes to the interaction histories. These provide a basis for categories that can be labeled with words. However, attributes can additionally become associated with the likelihood of specific outcomes. Each time an action-related interaction process succeeds or fails, the attributes of the target object are recorded, over time leading to statistics on the likelihood of success for future actions. For instance, heavy objects may slip from the

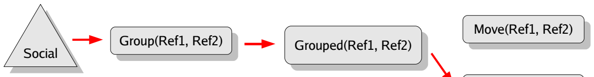

**Figure 3: A plan tree. The "social" motivation (triangle) is the root of the tree. The arrows denote a parent/child relation in the plan tree. The plan fragment to *Group* two objects has as its child the condition that the two objects be *Grouped*. The plan system selects between the options to *Move* one object towards the other, or vice versa, and the chosen action is executed.**

robot's fingers frequently, leading to a low success expectation for future lifting of heavy objects (sometimes we handpick the instances to be stored by the system, to prevent persistent sensorimotor failures from skewing success expectations). These statistics in turn affect the planning system's decision-making, and allow attributes to be directly linked to the affordances of each object, rather than merely being static data used for labeling purposes. Object schemas provide a discrete organization scheme for these statistics, which are then used by the planning system.

## 2.4 Planning System

The planning system governs which resource-bound interaction processes receive the resources to be able to activate, which mostly means coordinating motor activity for the robot. The planning system is built from, and makes decisions based on, the interaction processes and object schemas. This enables changes at the object schema level to rapidly influence planning.

*Plan trees* are hierarchical structures constructed starting with the three primary motivations of the system as roots. These primary motivations are: 1) to avoid damage (currently, just by avoiding collisions), 2) to address requests by the human partner (the "social" motivation), and 3) to explore objects by attempting to grasp and lift them (the "curiosity" motivation). Each of the primary motivations has a hand-coded *priority score* that is passed down to its children in the tree. The priority scores are set such that collisions will always be avoided with high priority, and exploration of objects only occurs when no requests or collisions need to be addressed.

Each non-root node of a plan tree is an interaction process, and thus the planning system is, like the belief context, another way to organize and view the interaction processes. Within a plan tree, plan fragment processes coordinate a sequence of other interaction processes (e.g., grasping an object by reaching for it and then closing the fingers). Condition processes monitor a specific condition (e.g., that the hand is holding a target object) and, when given priority, the planning system will search known plan fragment classes to instantiate an appropriate plan fragment process to satisfy a condition. Action processes are leaf nodes of the plan trees, and when active they send motor commands to the motor systems. Figure 3 depicts a plan tree.

As mentioned in the previous section, prior experience with objects and attributes enables the system to compile statistics on the likelihood of success for a given plan fragment or action process. When a condition process can be satisfied via multiple paths (such as in "Group the green block and the red apple"), the planning system selects the path with the highest likelihood of success.

Because the planning system is also an organized set of interaction processes, results and data from the interaction processes that constitute a plan tree have immediate effects on the plan tree. This can lead rapidly to interruption and revision of the plan tree as new information is assimilated into the object schemas.

## 2.5 Language System

Like the planning system, the language system is also built upon interaction processes and object schemas, enabling interpretations to shift with changes in sensorimotor context.

The system handles speech input in several stages. First, the speech recognizer converts the audio stream into a lattice of likely word tokens, which is then parsed by a probabilistic Earley parser (from [20]) and used to instantiate plans and actions in the main module of the system. Our system handles three kinds of linguistic input: descriptive, directive, and corrective. Descriptive inputs contain attribute information, such as "The red ball is heavy." Directive inputs communicate a desire from the human, such as "Pick up the red ball." Corrective inputs make changes to the immediately preceding directive input, as in "No... the green ball." Corrective inputs are converted to a corrected directive input and then treated as such.

In either type of verbal input, a phrase like "the red ball" presumably refers to an object in the shared domain of the human and robot. It must thus be matched to an object schema in the belief context so the verb phrase (such as "pick up" or "is heavy") can be processed. The matching of noun tokens to object schemas is handled by *reference processes*, which search the belief context for object schemas with corresponding attributes; "the red ball" leads to a reference process with the argument:

```
(refexpr (= function (lambda (x) (p_and
     (red x) (circle x)))) (= definite "definite")
                           (= cardinality "1"))))
```

This indicates a search for an object schema whose interaction history includes a color that matches the category for "red" and a shape that matches the category for "ball." Once found, the reference process connects the structures resulting from the verb phrase to its matching object schema.

In descriptive inputs, the verb phrase contains additional attribute information for the object schema. In the example "The red ball is heavy," once the reference process has matched an object schema for "the red ball," the categorical attribute corresponding to "heavy" is added to the interaction history of the weight-measuring process for that object schema.

In directive inputs, the verb phrase contains an action to be taken towards the referent object. In the example, "Touch the red ball," once the reference process has matched an object schema, a plan fragment process is added to a plan tree under the primary motivation for human interaction, so that the robot will move to and tap the appropriate object. The planning system then searches and assigns priorities to carry out the request.

Language output by the system currently consists of simple responses to identification questions such as "Describe the green one," to which it might say, "It is a small green block." Other output includes error responses such as "Sorry, I can't find it" and "Sorry, I can't do that."

# 3. RESULTS AND LIMITATIONS

Our intention is to build a system that demonstrates that sensorimotor responsiveness can be integrated with planning and language via interaction processes and object schemas. In this section, we discuss the behaviors that result and some of the limitations of the current implementation.

## 3.1 Resulting Behaviors

These are examples of the notable interactions that arise from the use of the system.
Non-verbal responsive behaviors:

- If forces above a set threshold are encountered by any of the finger sensors or the arm load cells, the robot pulls away from the direction of collision.
- If during a grasp action, the fingers encounter forces indicative of a successfully grasped object, then the targeted object is flagged as being grasped by the hand. For the duration of the grasp, the robot hand's position is then used to determine the location of the object.
- If the robot detects a successful grasp and moves the object, but then visually confirms (based on a distance metric in color, shape, and size) that contrary to expectations its targeted object is not moving, the object schema corrects its state to disregard the grasp-based location.
- When the robot has nothing to do, it will randomly grasp, lift, and put down objects on the table. As it does so, it observes their attributes and collects success statistics on each action relative to the object's attributes.

Verbal requests that the robot obeys:

- "Touch the green block."
- "Touch the red apple and the blue block."
- "Before touching the green block, touch the red ball."
- "Pick up the red ball."
- "Move the yellow block to the left."
- "Move the green block behind the yellow ball."
- "Group the green block and the red apple."

Responsive verbal behaviors:

- If the robot is responding to "Pick up the red ball," and the arm collides with the ball while reaching towards it, the system will detect the collision, pull the arm back, and revise its target location for the ball in order to make use of the collision information.
- If the robot is responding to "Move the yellow block to the left," and as the robot is lifting the block the human says, "No, to the right," the robot can interrupt its plan and move the block to a point to the right instead.
- If the robot is responding to "Touch the red block," and as the robot starts to move the human says, "No... the blue block," the robot can interrupt its plan and move towards the blue block instead.
- If the robot is responding to "Group the green block and the red apple" by moving the red apple, and the human mentions that "The red apple is heavy," the robot will interrupt the plan and move the green block instead.
- If the robot is responding to "Group the green block and the red apple," and after multiple attempts it fails to lift the red apple, it will move the green block towards the red apple instead.

## 3.2 Limitations

The design of the system enables a certain amount of robustness to sensorimotor noise and failure. The use of distance metric-based visual tracking enables the system to follow visual regions from frame to frame, despite occasional shadows and occlusion. Also, the compilation of success statistics allows the system to form and alter its plans based on its past experiences. Thus, if an object is consistently difficult to grasp, the system can choose alternative plans, as seen in the response to "Group the green block and the red apple."

However, sensorimotor noise still presents considerable difficulties for the system. Occasionally, an object's visual region looks different from its original form, or visual regions will split and merge briefly. Sometimes this leads to the creation of spurious object schemas, which can cause reference processes to match objects that seem to immediately vanish. At other times, the visual region of an object changes and the system believes the initial object has disappeared, leading to failure of all associated plans and actions. Also, shape and color recognition are often fragile to lighting conditions, and misrecognition frequently leads to poor matching for noun phrases.

Given our focus on building responsive linguistic behaviors, adding sensorimotor robustness is important but does not significantly detract from our primary intent to demonstrate the value of using an object schema-based model. Future work may include the further use of object schemas to add error-correction to the system amid noise in the sensorimotor modules. One key improvement would be to add conditions to the planner to take actions or revise beliefs based on the state of reference processes, which could permit retrying of reference matching and actions appropriately.

# 4. RELATED WORK

Our system is an effort to bring aspects of 1) behavior-based robotics into 2) language grounding systems by coordinating via 3) an affordance-centered, schema-based object representation. To our knowledge, the integration of facets from all three of these directions is novel. In this section we survey some of the relevant related work in the specific fields.

## 4.1 Behavior-Based Robotics

The importance of designing robots to react rapidly to their environment, with minimal modeling and internal state, is advocated in behavior-based robotics, notably Brooks' subsumption architecture [6, 7], which emphasizes the use of complete behavioral loops from inputs to outputs. Each of our interaction processes is designed in a similar fashion, by taking inputs, providing outputs, and looping constantly.

The integration of behavior-based reactivity with planning and action selection has been accomplished in models such as the three-layer architectures of Gat [17, 18] and Bryson [9, 10]. Like the three-layer architectures, our system could be viewed as a multi-layer model with reactive components at the bottom, a planning layer at the top, and an object schema layer that mediates between them. The specific use of object schemas to coordinate between reactivity and planning is novel, and allows for our system's unique language and representational capabilities.

Projects done by Blumberg et al. [11, 22] and later Breazeal et al. [4, 5] are fairly reactive while using distance metrics to compile object beliefs over time and later using hierarchical plan representations as well. However, the usage of speech, while sometimes present, is very limited, and much of the object processing is decoupled from the object representations themselves, making it inconvenient to represent object-directed affordances such as "liftable" and "graspable."

## 4.2 Language Grounding Systems

Numerous projects exist which ground language to the real world, via vision, simulation, and robots (see [27] or [28] for surveys of these). Notably, Winograd's SHRDLU [35] carried out natural language commands in a symbolic "simulator" of a blocks-world manipulation robot, but which ignored the complexity of connecting to the real sensorimotor world. The richness of the sensorimotor connection, and the need for responsiveness in a real environment, is a key feature of our system, and necessary for compelling language use in a real-world robot.

Language-grounding systems such as [1, 26, 30] typically connect aspects of language to features in a single frame of input, or assume that objects are reasonably static and can be identified from frame to frame. In contrast, our system assigns dynamic tracking processes to objects and explicitly handles changing information about objects.

In works such as [2, 23, 31], language is used to give commands to mobile robots, typically in the form of actions relative to landmarks. Due to our focus on manipulation rather than navigation, our emphasis is on the structure of interactions with objects and associated language.

Several other researchers pairing language and robotics [12, 32, 34] focus on evolving novel languages between agents or robots trying to communicate. Our focus is on representations specifically appropriate to pre-existing human languages.

In our group's Ripley robot system [21, 29], language was used to issue commands such as "Hand me the blue one," but action failures had no impact on decision-making beyond a simple retry mechanism. Our new system specifically uses a planner with statistics on action successes to make decisions and replan dynamically.

An additional novel benefit of our approach is that it provides a clear foundation for grounding words like "liftable" or "graspable," because it organizes actions into sets of affordances centered on objects.

## 4.3 Affordance-Centered and Object Schema Models

A number of systems learn about object affordances [15, 16, 24, 33]. Learning is a vital future direction for our work, but our current emphasis is on connecting an affordance-centered representation to the responsiveness necessary for performing tasks at the human's request.

An addition that our system makes to the affordance-centered view is the use of seemingly-static attributes, such as color, shape, and weight, as a source of information for affordance expectations. For instance, heavy objects ("heavy" being an attribute) are generally more difficult to lift ("liftable" being an affordance), and the system will notice this over multiple experiences, leading the robot to seek an alternative to lifting a heavy object.

Other models focus on objects modeled as schemas in terms of actions and perceptions: Drescher [14] with a microworld implementation, Roy [28] in his theoretical schema framework, and Gorniak and Roy [20] in a video game environment. In treating objects as nothing more than the sum of their affordances, this approach is also in line with the schema models.

Like object schema models or affordance-based approaches, our approach treats objects as a set of actions and expectations, at the level of the interaction processes. The object schemas then act as discrete entities for language and planning. This allows our system to operate in a continuous sensorimotor environment while using language and planning, a combination not present in the other works.

# 5. SUMMARY AND FUTURE DIRECTIONS

We have implemented a robot system that uses object schemas to combine aspects of behavior-based robotics and language-grounding systems. The use of concurrent interaction processes at all levels of the system enables responsiveness to environmental changes and new verbal inputs. The organization of interaction processes and their histories into object schemas provides a convenient affordance-based representation for sharing information between related processes. The object schemas also provide discrete entities for use in the planning and language systems.

Brooks' work on the subsumption architecture [6, 7] emphasized the need for robot systems to be designed around complete behavioral loops, treating responsiveness to the environment as a key element of robot design. As interactive robots increasingly make use of natural language, we similarly believe that language use should also be tightly coupled to the sensorimotor environment. By building our object schemas out of interaction processes, we believe our system incorporates that premise while allowing for planning and language use.

## Acknowledgments

This paper is partly based upon work supported under a National Science Foundation Graduate Research Fellowship. Many thanks to Jonathan Huggins, Thananat Jitapunkul, Peter Lai, John McBean, Kailas Narendran, Philipp Robbel, David Wang, and Jonathan Wang, for their time and effort with numerous aspects of the robot system.

# 6. REFERENCES

[1] C. Bauckhage, J. Fritsch, K. Rohlfing, S. Wachsmuth, and G. Sagerer. Evaluating integrated speech and image understanding. In *Proceedings of the IEEE International Conference on Multimodal Interfaces (ICMI)*, pages 9–14, 2002.

[2] P. Beeson, M. MacMahon, J. Modayil, A. Murarka, B. Kuipers, and B. Stankiewicz. Integrating multiple representations of spatial knowledge for mapping, navigation, and communication. In *Proceedings of the AAAI 2007 Spring Symposium on Control Mechanisms*

*for Spatial Knowledge Processing in Cognitive / Intelligent Systems*, 2007.

[3] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(24):509–522, 2002.

[4] M. Berlin, J. Gray, A. L. Thomaz, and C. Breazeal. Perspective taking: An organizing principle for learning in human-robot interaction. In *Proceedings of AAAI 2006*, 2006.

[5] C. Breazeal, M. Berlin, A. Brooks, J. Gray, and A. L. Thomaz. Using perspective taking to learn from ambiguous demonstrations. *Journal of Robotics and Autonomous Systems Special Issue on Robot Programming by Demonstration*, 54(5), 2006.

[6] R. A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2:14–23, 1986.

[7] R. A. Brooks. Intelligence without representation. *Artificial Intelligence*, 47:139–160, 1991.

[8] J. Bruce, T. Balch, and M. Veloso. Fast and inexpensive color image segmentation for interactive robots. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2000.

[9] J. J. Bryson. *Intelligence by Design: Principles of Modularity and Coordination for Engineering Complex Adaptive Agents*. PhD thesis, MIT, Department of EECS, Cambridge, MA, June 2001. AI Technical Report 2001-003.

[10] J. J. Bryson and L. A. Stein. Modularity and design in reactive intelligence. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 1115–1120, August 2001.

[11] R. Burke, D. Isla, M. Downie, Y. Ivanov, and B. Blumberg. Creature smarts: The art and architecture of a virtual brain. In *Proceedings of the Game Developers Conference*, pages 147–166, 2001.

[12] A. Cangelosi. The grounding and sharing of symbols. *Pragmatics and Cognition*, 14:275–285, 2006.

[13] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5), May 2002.

[14] G. Drescher. *Made-Up Minds: A Constructivist Approach to Artificial Intelligence*. MIT Press, 1991.

[15] P. Fitzpatrick. *From First Contact to Close Encounters: A Developmentally Deep Perceptual System for a Humanoid Robot*. PhD thesis, Massachusetts Institute of Technology, 2003.

[16] P. Fitzpatrick and G. Metta. Grounding vision through experimental manipulation. *Philosophical Transactions of the Royal Society: Mathematical, Physical, and Engineering Sciences*, 361(1811):2165–2185, 2003.

[17] E. Gat. Integrating planning and reaction in a heterogeneous asynchronous architecture for controlling mobile robots. In *Proceedings of the Tenth National Conference on Artificial Intelligence (AAAI)*, 1992.

[18] E. Gat. Three-layer architectures. In D. Krotenkamp, R. P. Bannasso, and R. Murphy, editors, *Artificial Intelligence and Mobile Robots*. AAAI Press, 1998.

[19] J. J. Gibson. *The Ecological Approach to Visual Perception*. Erlbaum, 1979.

[20] P. Gorniak and D. Roy. Situated language understanding as filtering perceived affordances. *Cognitive Science*, 31(2):197–231, 2007.

[21] K. Hsiao and D. Roy. A habit system for an interactive robot. In *AAAI Fall Symposium: From Reactive to Anticipatory Cognitive Embodied Systems*, 2005.

[22] D. Isla, R. Burke, M. Downie, and B. Blumberg. A layered brain architecture for synthetic creatures. In *Proceedings of IJCAI*, 2001.

[23] L. S. Lopes and J. H. Connell. Semisentient robots: Routes to integrated intelligence. *IEEE Intelligent Systems*, 16:10–14, 2001.

[24] J. Modayil and B. Kuipers. Where do actions come from? Autonomous robot learning of objects and actions. In *Proceedings of the AAAI 2007 Spring Symposium on Control Mechanisms for Spatial Knowledge Processing in Cognitive / Intelligent Systems*, 2007.

[25] J. Piaget. *The Construction of Reality in the Child*. Basic Books, 1955.

[26] T. Regier and L. Carlson. Grounding spatial language in perception: An empirical and computational investigation. *Journal of Experimental Psychology*, 130(2):273–298, 2001.

[27] D. Roy. Grounding words in perception and action: computational insights. *Trends in Cognitive Science*, 9(8):389–396, 2005.

[28] D. Roy. Semiotic schemas: A framework for grounding language in action and perception. *Artificial Intelligence*, 167(1-2):170–205, 2005.

[29] D. Roy, K. Hsiao, and N. Mavridis. Mental imagery for a conversational robot. *IEEE Transactions on Systems, Man, and Cybernetics*, 34:1374–1383, 2004.

[30] J. M. Siskind. Grounding the lexical semantics of verbs in visual perception using force dynamics and event logic. *Journal of Artificial Intelligence Research*, 15:31–90, August 2001.

[31] M. Skubic, D. Perzanowski, S. Blisard, A. Schultz, W. Adams, M. Bugajska, and D. Brock. Spatial language for human-robot dialogs. *IEEE Transactions on SMC Part C, Special Issue on Human-Robot Interaction*, 34(2):154–167, May 2004.

[32] L. Steels and T. Belpaeme. Coordinating perceptually grounded categories through language: A case study for colour. *Behavioral and Brain Sciences*, 28:469–529, 2005.

[33] A. Stoytchev. Behavior-grounded representation of tool affordances. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2005.

[34] P. Vogt and F. Divina. Social symbol grounding and language evolution. *Interaction Studies*, 8(1):31–52, 2007.

[35] T. Winograd. *Understanding Natural Language*. Academic Press, 1972.