

PSchema: A developmental schema learning framework for embodied agents

Michael Sheldon

Department of Computer Science
Aberystwyth University
Aberystwyth, Ceredigion, SY23 3ET, UK
Email: mts08@aber.ac.uk

Mark Lee

Department of Computer Science
Aberystwyth University
Aberystwyth, Ceredigion, SY23 3ET, UK
Email: mhl@aber.ac.uk

Abstract—In this paper we introduce PSchema, a framework for Piagetian schema learning which allows for the direct use of symbolic schema learning in a robotic environment. We show the benefit of a developmental progression to aid in the learning of the system and introduce a generalisation mechanism which further increases the capabilities of these techniques. Using a robotic arm we demonstrate the system’s ability to learn to touch objects placed in front of it and how it can represent the knowledge gained from this in a manner suitable for continuous on-line learning. We then go on to demonstrate how these mechanisms can be used to provide a framework for the learning of language, grounded in the robot’s sensory perception of the world.

Index Terms—Embodied Cognition, Language Acquisition, Grounding of Knowledge and Representations, Developmental Learning

I. INTRODUCTION

Drescher [1] suggested a constructivist approach to learning based on Piagetian ideas using the notion of ‘schemas’. Schemas are units of knowledge associating perceptions, actions and predictions. If the environment is perceived to be in a certain state then taking an action associated with this state should cause the environment to change to match the perceptions anticipated by that schema’s predictions.

In its simplest form a schema consists of a set of pre-conditions, an action and a set of post-conditions (often represented in the form *pre-conditions/action/post-conditions*), providing a basic forward learning model. These schemas can then be chained by connecting the post-conditions and pre-conditions of different schemas together to create a traversable network representing different world states and the actions required to move between them.

Holmes and Isbell [4] extended Drescher’s work to enable the use of continuous value sensors (the original implementation was limited to binary sensors). They showed that it was possible to model Partially Observable Markov Decision Processes (POMDPs) via this mechanism.

Perotto, et al. [10] introduce a Constructivist Anticipatory Learning Mechanism (CALM), which makes use of a schema based learning mechanism. The schemas are organised in a tree hierarchy going from most general to most specific, making it possible for the system to fall back on more general solutions if a specific one fails or is unavailable. In contrast to Holmes and Isbell this system took a property based approach

providing a more direct mapping between the environment and the agent’s perceptions than a state based representation. The generalisation mechanism proposed relied primarily on determining which properties could be ignored in a given context; by contrast the generalisation mechanism we describe in section II-E constructs more expressive hypotheses as to how the robot’s perceptions relate to one another.

Guerin and McKenzie [2] have since used schema learning in a simple simulated robotic environment, but as yet little work has been performed using this technique on a physical robot. They also introduced the concept of superschemas where multiple schemas can contribute values towards a target action, this allows the system to combine different classes of actions to provide new behaviours.

Oudeyer and Kaplan [9] explore the intrinsic motivation of language learning rooted in play and curiosity, showing how an intrinsic motivation system can allow a robot to self-organise its learning process.

Hart [3] applies a developmental approach to an intrinsically motivated robotic system targeting the learning of visual and motor skills and considers how these can be learnt in a generalised form.

The system we describe follows a developmental progression, the later stages of which are modelled on the work of Iverson and Goldin-Meadow [5], consisting of the following stages:

- Motor babbling
- Motor vision mapping
- Failed grasping leading to proto-imperative pointing
- Complementary one word speech with pointing
- Supplementary one word speech with pointing
- Two word sentences

Previously [11] we discussed these developmental stages in detail and described the progression to stage three, leading to a robotic system capable of learning to communicate in the form of simple pointing gestures based around a schema learning architecture. In this paper we detail the underlying schema mechanisms that support this progression, and extend it to encompass early language learning.

II. SCHEMA LEARNING

In the following we highlight the key features of the PSchema framework and describe in detail the advances offered by this system.

A. Observation probability tracking

In addition to tracking the probability of a schema's success as a whole, PSchema tracks the probability of each individual observation within that schema. This means that when a chain of schemas is sought after to complete a given task only the relevant components are considered. For example, if the robot has been given the task of moving a block but one of the potential schemas that could be used to complete this task also has a chance of knocking a ball off the table in the process, the likelihood of the ball being displaced can be ignored as it is not relevant to the completion of the task.

Tracking individual probabilities also allows the system to cope with sensor noise to a greater degree. Instead of creating a new schema on the few occasions when sensor noise has resulted in a different outcome to that expected the system can store this alternative outcome alongside the expected result with the appropriate probability for each.

B. Associated observations

Previous schema systems have tracked the pre-conditions necessary for a schema to be successful and the post-conditions which should occur after the schema has been executed. In addition to this we introduce the concept of 'associated observations'. These are observations that have been seen to occur frequently alongside a schema but are neither required for the schema to be executed, nor directly effected by the action taken. This provides the basis for the introduction of language into the system, without the need for any explicit concept of language being preprogrammed into the system. The process by which this takes place is discussed alongside the language learning results in section VI-B.

C. Schema excitation

To determine which schema should be executed next we make use of an intrinsic motivation system, focusing on the novelty of experiences [9]. When presented with a novel scenario this leads to executing schemas which are likely to be relevant to the novel aspects of the scenario and so more likely to lead to the formation of new schemas representing the effects of the novel components of the scenario. This gives the system the ability to form partial plans of action [13] aimed at expanding its own knowledge of the world.

A schema's excitation level is found by first comparing each observation present in the current world state (Υ) with all the pre-conditions (Ψ) and associated observations (α) of that schema, with associated observations being weighted to have less impact than pre-conditions, in our experiments this weighting (ω) is set to 0.8. This weighting makes it possible for the primary sensations directly linked to the executability of a schema to take precedence over the potentially less

relevant associated observations in the early stages of learning when primary and associated sensations may have been observed a similar number of times.

Each observation contains a set of different properties, the amount an observation remembered as part of a schema is excited by an observation currently present in the environment is determined by how many of these properties are the same. For example a simple visual observation may have properties specifying in which visual field an object is detected and the colour of that object. This allows the observation of a blue block in field 7 to excite an observation of the robot's own green end effector (a touch sensitive 'finger') in that same field. As such, although the robot has never encountered the block before it is directed towards schemas that are most likely to have some relation to it.

The excitation contribution of each observation is then weighted based on the amount that observation has been encountered in the past, with more common observations being less interesting than novel ones. To do this the system tracks the number of times an observation is given attention ($N(x)$, where x is an observation). An observation is considered to have been given attention when it is both being perceived by the robot and is also referenced in the currently executing schema. In this way the importance of a perception not directly related to the current action is not diminished unnecessarily. For example if the robot is presented with two objects, one which has been previously seen and one which is new, the new object will be of more interest and so will be interacted with, however although the old object is constantly being perceived during these interactions the number of encounters with it is not increased. As such the level of excitement provided by that object remains unchanged while it is not being interacted with.

If a schema cannot be activated directly from the current state but instead requires a chain of preceding actions we decrease the excitation of that schema based on the distance (d) between the current world state and that schema, this distance is defined as being the length of the chain of schemas required to achieve the schema currently being evaluated.

The overall formula for excitation can be expressed as:

$$E(\{\Psi, \alpha\}|\Upsilon) = \begin{cases} 0 & \text{if unreachable,} \\ \frac{1}{d} \left(\sum_{i=0}^{i=|\Upsilon|} \sum_{j=0}^{j=|\Psi|} \frac{|\Upsilon_i \cap \Psi_j|}{N(\Psi_j)} \right. \\ \left. + \omega \sum_{i=0}^{i=|\Upsilon|} \sum_{j=0}^{j=|\alpha|} \frac{|\Upsilon_i \cap \alpha_j|}{N(\alpha_j)} \right) & \text{otherwise.} \end{cases} \quad (1)$$

A schema is considered unreachable if no chain of previously learned schemas can be formed to transition from the current world state to one in which that schema can be executed.

The schema with the highest excitation value is then selected for execution.

D. Schema creation

Prior to schema creation an existing schema must have been executed. This schema is selected based on the excitation criteria outlined above and so is likely to be the most relevant action in that context, as it will be the schema with the highest number of uncommon observations that can still be satisfied by the current world state.

To decide if a new schema should be created we first take the relative complement of the current world-state (after schema execution) with respect to the world-state prior to execution plus the predicted post-conditions. If the result of this is anything other than the empty set then an unexpected outcome has occurred.

If it is found that a new outcome has occurred in conjunction with a new observation being encountered prior to the execution of the schema then a new schema is created to represent this knowledge. If the observations present prior to the execution of the schema are the same as the pre-conditions to the schema then the new outcome is added to an existing schema and the probability of it occurring is tracked. An illustration of this process can be seen in figure 1.

E. Schema generalisation

Schema generalisation allows the system to go beyond simply being able to predict and form action plans based around previously experienced outcomes, giving it the ability to make informed decisions about scenarios it hasn't encountered yet but which are similar to past experiences.

Generalisation is attempted whenever a new schema is created. The generalisation process first selects the subset of schemas which appear to be similar to the new schema based upon them all having the same number of the same type of observations for their pre-conditions and post-conditions. At this time associated observations are ignored for the process of generalisation, but observations can be associated with existing generalised schemas.

To make it possible to generalise the action component of the schema we must first be able to describe it in terms of observations. We achieve this by finding the result of that action in the simplest known context. The simplest context is discovered by finding a schema which makes use of that action and has the least number of pre-conditions, all of which must be satisfied by the pre-conditions in the schema currently being generalised over. The action is then converted into a 'target action' which consists of a list of observations that should be achieved by any schema implementing that action. An example of this process can be seen in figure 2.

Once the schema is in a form entirely represented by observations a simple lifting process takes place, replacing any identical values that occur in the pre-conditions and in either the target action, the post-conditions or both with a randomly generated variable (represented within our system as \$x where x is any alphabetic character). An example of the conversion from a concrete schema to a generalised schema can be seen in figure 3.

Given the world state:

World state
Object in field 4

The following schema is selected, due to the visual observation of an object in field 4 triggering excitation of any schemas related to observations referencing field 4:

Pre-conditions	Action	Post-conditions
	Move to joint positions 0.43, 0.84	Finger in field 4

This schema is then executed and the process for determining if a new schema is required is performed:

World state post-execution
Object in field 4 Finger in field 4 Touching

World state pre-execution \cup Predicted post-conditions
Object in field 4 Finger in field 4



Relative complement
Touching

As this is not the empty set a new schema will be formed:

Pre-conditions	Action	Post-conditions
Object in field 4	Move to joint positions 0.43, 0.84	Object in field 4 Finger in field 4 Touching

Fig. 1. An example of the process leading to a new schema being created.

This generalised schema is then tested against all of the similar schemas that were found in the first stage of the process. If enough of these are correctly represented by the generalised schema it is added to the schema memory (this threshold is set at 75% for the experiments below, no optimisation of this value has yet been attempted).

When a generalised schema is executed the values from the current world state are used to populate the variables within the generalised schema, allowing it to be treated as a normal schema by all other aspects of the system.

F. Developmental control

The system implements a Lift Constraint, Act, Saturate (LCAS) [7], [6] loop to artificially constrain the inputs to the robotic system and so reduce the complexity of the learning required at each stage of the system's development. Constraints are placed upon the system's sensory input and the system then operates in this mode until there is little novel

Given the following schema as a potential target for generalisation.:

Pre-conditions	Action	Post-conditions
Object in field 4	Move to joint positions 0.43, 0.84	Object in field 4 Finger in field 4 Touching

We select the following schema based on it sharing the same action component and having the least number of pre-conditions. In this example the selected schema has no pre-conditions indicating that it is applicable in any context.

Pre-conditions	Action	Post-conditions
	Move to joint positions 0.43, 0.84	Finger in field 4



Pre-conditions	Action	Post-conditions
Object in field 4	Finger in field 4	Object in field 4 Finger in field 4 Touching

The post-condition of that schema is then used as a target condition to be achieved in place of the original concrete action. Upon execution of this action the schema most likely to achieve the target will be found and executed.

Fig. 2. An illustration of the process for forming a target action.

Pre-conditions	Action	Post-conditions
Object in field 4	Finger in field 4	Object in field 4 Finger in field 4 Touching



Pre-conditions	Action	Post-conditions
Object in field \$x	Finger in field \$x	Object in field \$x Finger in field \$x Touching

Fig. 3. A schema with its concrete action replaced by a target action can then be converted into a generalised schema.

input being found. A constraint is then lifted, allowing the system to build upon its knowledge from the previous stage whilst being exposed to a more complex and detailed view of the world. In addition to this we simplify the environment that the robot is initially exposed to, not introducing other objects for it to interact with until it has had the opportunity to learn how its own systems function and effect its senses, an approach similar to the scaffolding [8] performed by parents when helping children to learn.

In the first stages of learning the robot learns about its own body and the effects that its movements can have on its perceptions. After the robot has developed a suitable

representation of this we introduce coloured blocks for the robot to interact with, learning how it can interact with these in different locations and the ways in which these objects can cause different sensations for the robot. Finally we provide the robot with auditory input, speaking to it as it performs actions and allowing it to learn the relationship between these words and its own behaviours.

G. Habituation

It is important to allow the agent to habituate between developmental stages, this gives the system the opportunity to learn the different possible outcomes of any schemas that might not be 100% reliable (for example, due to sensor noise or poor repeatability of motor actions in the hardware platform). Without this the system may falsely attribute the sensory responses it receives that differ from the expected outcomes as being caused by an unrelated observation introduced during the later learning stages.

III. EXPERIMENTAL CONFIGURATION

A. Physical robot

The hardware that the system is being tested on consists of an Adept manipulator arm mounted on a rigid vertical back-plane. The arm is configured to operate on a two-dimensional manifold above a table upon which objects can be placed for it to interact with, the manifold curves up at the extremities tracing the outer limit of the robot's work envelope allowing for pointing towards distant objects. The arm has a single 'finger' as an end effector, which has four touch sensors attached giving directional touch input. This end effector can be used for interacting with objects by touching them and pushing them around the work area and for communicating by pointing at an object.

The vision system consists of an AVT Stingray F-046C firewire camera, which provides a resolution of 780x580 at up to 61 frames per second. This is mounted on a pan tilt platform above the arm looking down on the work space. The system's visual space is divided into a number of small circular visual fields, making the identification of object positions within the world more discrete. Objects are detected through simple blob detection and are identified based on their colour.

This hardware setup can be seen in figure 4.

B. Simulated robot

Due to the large running times of some of the experimental scenarios these have been tested in a simulation environment that has been constructed to roughly model the physical hardware. It is important to note that the scenarios requiring simulation are designed to illustrate the benefits of specific components within the system by their removal. In the scenarios in which the complete system is active a truly embodied approach with the previously described physical robot is employed.

In addition to the arm the environment contains a pan/tilt vision system, a touch sensitive end effector and a workspace on which objects can be placed. The simulator provides rigid



Fig. 4. The current hardware configuration.

body physics, allowing for semi-realistic interactions between the arm and its environment. This simulation environment can be seen in figure 5. The control software is capable of driving either the simulated arm or the real arm without modification. The simulator in use is Gazebo, a part of the Player project.

IV. EXPERIMENT 1: COMPARISON OF PERFORMANCE WITH AND WITHOUT GENERALISATION AND STAGED LEARNING

The aim in each of the following scenarios is for the robot to learn to touch an object placed at any location inside its working area or point to an object if placed outside of the working area.

Scenarios 2 and 3 exist to highlight the effects of the generalisation and developmental progression by their removal. They are not intended as an example of the system as a whole, but rather to show that without these features the approach would be too complex for real robotics, however with these techniques a suitable representation can be achieved quickly and in a small number of schemas, as demonstrated in the first scenario.

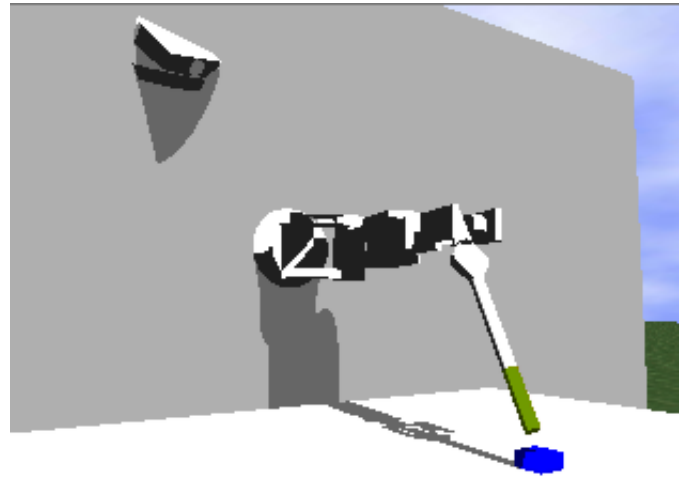


Fig. 5. The simulation environment showing the arm pointing at an object placed slightly outside of the robot's work envelope.

A. Scenario 1: Staged learning with generalisation

In this scenario the robot is given the opportunity to first learn how the movement of its arm can effect its visual perception of the world. After this a small blue block is introduced and the excitation this causes should result in the robot reaching towards it. Upon contact with the object the robot will receive a signal from its touch sensor. The object will then be moved into two or three further positions on the table, the expectation being that the robot will be able to generalise these few examples to represent touching the object anywhere on the table. Once a generalised schema representing this is created the object will then be moved in to a position that the robot cannot reach, however in attempting to touch the object it will form a pointing motion [11], [12] but will not receive a direct touch sensation, providing a counter example in which the generalised solution does not hold. In any cases where counter examples exist that contradict generalised solutions these are selected instead, allowing the system to form basic boundaries around generalised schemas.

This scenario has been performed both on the real robot and within the simulator, to show that the techniques outlined here translate across to usage on real systems.

B. Scenario 2: Staged learning without generalisation

As in scenario 1 the robot is first allowed to learn the visual changes caused by the movement of its end effector, after which an object is introduced. However, unlike the previous example the system's ability to generalise from past experiences is disabled. As a result, to form an equivalent representation of the world the object must be placed in each visually distinct location upon the table.

Due to the requirement to place the object in each location on the table this scenario was only performed in the simulator where this activity could be automated, greatly reducing the experimentation time.

C. Scenario 3: Learning without stages, with generalisation

In this scenario the opportunity to learn about the effects of moving its manipulator prior to interaction with objects is denied to the robot.

As this scenario required thousands of actions to take place, in addition to the requirement from scenario 2 in which the object must be repositioned many times this scenario was also only performed in simulation.

V. EXPERIMENT 2: LEARNING AND RESPONDING TO LINGUISTIC COMMANDS

The system receives linguistic input through the use of speech recognition software, this converts the simple single word utterances to text tokens which are then passed on to the schema learning system.

For this experiment the system starts in the end condition of experiment 1, scenario 1, having learnt a generalised schema representing touching. An object is then placed in a previously untested position to ensure that it is exciting enough for the robot to reach for immediately. When the robot reaches for the object a human operator says the word ‘touch’. The robot is then left to ‘play’ with the object until it loses interest and begins to execute other unrelated schemas. The operator then says the word ‘touch’ again, and the robot’s attention should be directed back to the object.

To confirm that this word has been associated with a generalised mechanism for touching the block is then placed in another previously untested location. The operator once again waits until the robot is no longer interested in the object and then says the word ‘touch’, as before the robot should then attempt to touch the object.

VI. RESULTS

A. Experiment 1

Scenario	Schemas produced
Scenario 1 (Physical Robot)	115
Scenario 1 (Simulated Robot)	227
Scenario 2 (Simulated Robot)	347
Scenario 3 (Simulated Robot)	19244

Scenario	Object Placements
Scenario 1 (Physical Robot)	2
Scenario 1 (Simulated Robot)	2
Scenario 2 (Simulated Robot)	100
Scenario 3 (Simulated Robot)	100

The difference in figures for the physical and simulated robot in scenario 1 is due to the differences in the visual properties of the two systems. The simulated robot has a much wider field of view, resulting in a greater number of visual fields.

It is important to note that while the difference between scenarios 1 and 2 may not be that great in terms of the number

of schemas created, a roughly similar amount of additional schemas would need to be added for every new object encountered by the system due to the lack of generalisation in scenario 2. So while scenario 3 has a far greater number of schemas, arguably it can represent the robot’s possible interactions with the world more completely as it can generalise to different objects without requiring object-specific learning. Additionally the number of object placements required to train the system in scenario 2 is much higher as without generalisation the object must be seen in each position on the table to build an equivalent representation of object touching, whereas in scenario 1 only 2 examples are required before the system is able to generate a valid generalisation.

The large number of schemas and actions required to form a complete representation in scenario 3 are a result of the robot not being given the opportunity to learn about the effects of its actions in a simpler context. As such it incorrectly considers the presence of an object in a particular field to be a pre-condition of any possible action (it has never experienced these actions without an object present). While our chosen mechanism for avoiding this problem is the use of a series of learning stages, gradually increasing in complexity, an alternative solution to this problem might be to make use of a more complex saliency filter to make additional assumptions about what may or may-not constitute a pre-condition. However we believe our staged learning approach offers a more flexible solution as it allows the system to be trained in a variety of environments, rather than pre-programming it with assumptions about the world in advance.

It is worth noting that even when operating with close to 20,000 schemas in scenario 3 the system was still capable of functioning in real-time.

B. Experiment 2

Figure 6 shows a number of labelled peaks highlighting key points within the experiment. Peak (a) is the point at which the object is first introduced, along with the first utterance of the word ‘touch’. The excitation caused by seeing the object causes the robot to begin interacting with it. After this excitation decreases and the robot begins executing schemas unrelated to the object. Peak (b) shows the excitation increasing again when the word ‘touch’ is heard for a second time, activating the associated touching schema and directing the robot’s attention back to the object. At line (c) the object is moved into a new position, without any linguistic input. Finally peak (d) is the robot hearing the word ‘touch’ again and being directed back to touching the object, now in a new position.

As mentioned in the section on associated observations the interactions between the associated observations and the excitation system can result in some interesting effects when it comes to attempting to teach the system to respond to spoken instructions. As can be seen from these results it is only necessary to give a small number of examples for a word to be potentially used as a command to direct the robot back to the action being performed at that time.

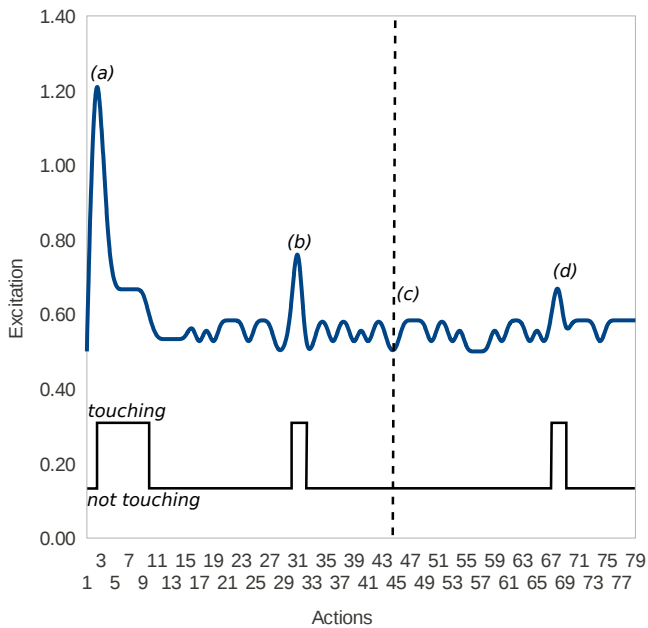


Fig. 6. *Top*: Level of excitement provided by the most excited schema at each time-step during the language experiments. *Bottom*: The type of schema being executed by the system, with the high state representing a touching schema and the low state being any other schema.

In these experiments we allow the excitation from the auditory sensations to decay at the same rate as any other sensation. This means that if the same word is repeated often enough the robot will temporarily find it less exciting than other actions, once these actions have been performed (so lowering their excitation) the word will once again be exciting enough to trigger the related action. The primary aim of our system is to direct attention towards actions likely to result in new learning experiences, not to respond to commands. If a command driven system was desired the excitation from auditory input could simply be excluded from the decay applied to other forms of sensory input.

VII. CONCLUSIONS

The results presented show a clear advantage for the use of a staged developmental progression when applying schema learning to robotics in this manner. While it was possible to learn the same representation without a staged learning approach, the number of actions required would make this highly impractical outside of simulation or without a saliency filter, which would be likely to introduce additional assumptions about the world. The generalisation mechanism further reduces the number of actions required to learn the scenario and the number of schemas necessary to represent it.

The addition of the generalisation mechanism and the concept of associated observations makes simple verb based language learning possible. Without the generalisation mechanism a word would need to be relearned for each instance of an action

in different contexts, and without the associated observations language could only be represented as pre-conditions of an action, meaning that the word would have to be heard before that action could be carried out.

VIII. FURTHER WORK

The linguistic aspects investigated here only deal with verbs, which map fairly directly on to entire schemas. Future work will look at ways in which nouns and adjectives may be associated with observations or groups of observations separate from specific schemas, utilising a mechanism similar to the schema generalisation presented here for associating words with related components of observations. This will allow for linguistic input in the form of two word sentences comprising noun-verb pairs to direct action more precisely.

While the results show an ability to respond to commands after a single example, the system isn't exposed to much linguistic noise that could cause confusion as to the correct associations. The probability tracking system should allow for this to be overcome in noisier environments, but this has yet to be comprehensively tested.

IX. ACKNOWLEDGEMENTS

This work is funded by EC-FP7, ICT – 216125, an investigation into the emergence of communication in Robots through Sensorimotor and Social Interaction (ROSSI).

REFERENCES

- [1] G. L. Drescher. *Made-Up Minds: A Constructivist Approach to Artificial Intelligence*. MIT Press, 1991.
- [2] F. Guerin and D. McKenzie. A Piagetian model of early sensorimotor development. *Proceedings from Epigenetic Robotics 2008*, 2008.
- [3] S. W. Hart. The development of hierarchical knowledge in robot systems. 2009.
- [4] M. P. Holmes and C. L. Isbell Jr. Schema Learning: Experience-Based Construction of Predictive Action Models. *Advances in Neural Information Processing Systems*, 17:585–592, 2005.
- [5] J. M. Iverson and S. Goldin-Meadow. Gesture Paves the Way for Language Development. *Psychological Science*, 16(5):367–371, 2005.
- [6] M. H. Lee, Q. Meng, and F. Chao. Developmental learning for autonomous robots. *Robotics and Autonomous Systems*, 55(9):750–759, 2007.
- [7] M. H. Lee, Q. Meng, and F. Chao. Staged competence learning in developmental robotics. *Adaptive Behavior*, 15(3):241, 2007.
- [8] M. Lungarella, G. Metta, R. Pfeifer, and G. Sandini. Developmental robotics: a survey. *Connection Science*, 15(4):151–190, 2003.
- [9] P. Y. Oudeyer and F. Kaplan. Discovering communication. *Connection Science*, 18(2):189–206, 2006.
- [10] F. Perotto, J. Buisson, and L. Alvares. Constructivist anticipatory learning mechanism (CALM) - dealing with partially deterministic and partially observable environments. *Proceedings of Epigenetic Robotics 2007*, 2007.
- [11] M. Sheldon and M. Lee. A developmental approach to the emergence of communication in socially situated embodied agents. In *Proceedings of the Ninth Annual IEEE International Conference on Development and Learning*, pages 204–210. IEEE, 2010.
- [12] L. S. Vygotsky. *Mind In Society: Development of Higher Psychological Processes*. page 56. Harvard University Press, 1978.
- [13] L. S. Zettlemoyer, H. M. Pasula, and L. P. Kaelbling. Learning planning rules in noisy stochastic worlds. In *Proceedings of the National Conference on Artificial Intelligence*, volume 20, page 911. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2005.