Intrinsically motivated developmental learning of communication in robotic agents

Michael Sheldon

Department of Computer Science

Aberystwyth University

Declaration

This work has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

Signed: _____

Date: _____

Statement 1

This thesis is the result of my own investigations, except where otherwise stated. Where correction services have been used, the extent and nature of the correction is clearly marked in a footnote(s).

Other sources are acknowledged by footnotes giving explicit references. A bibliography is appended.

Signed: _____

Date: _____

Statement 2

I hereby give consent for my thesis, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed: _____

Date: _____

Dedicated to our future robot overlords.

Acknowledgements

Firstly, I'd like to thank my supervisor, Mark Lee, whose support and insight have been invaluable in completing this work.

A big thank you must also go out to all my colleagues in both the ROSSI and IM-CLeVeR projects: Tao Geng, Martin Hülse, James Law, Patricia Shaw and James Wilson, who have all been a pleasure to work with. As too have all the members of the Computer Science Department that I've had the opportunity to work alongside on various smaller projects.

I'd also like to thank my parents for their support in general and for helping to proof read much of this thesis.

Finally, a thank you to all the various, and varied, denizens of Aberystwyth, who have helped to make my time here as interesting and odd as it has been informative.

Abstract

This thesis is concerned with the emergence of communication in artificial agents as an integrated part of a more general developmental progression. We demonstrate how early gestural communication can emerge out of sensorimotor exploration before moving on to linguistic communication. We then show how communicative abilities can feed back into more general motor learning.

We take a cumulative developmental approach, with two different robotic platforms undergoing a series of psychologically inspired developmental stages. These begin with the robot learning about its own body's capabilities and limitations, then on to object interaction, the learning of proto-imperative pointing and early language learning. Finally this culminates in more complex object interaction in the form of learning to build stacks of objects with the linguistic capabilities developed earlier being used to help guide the robot's learning.

This developmental progression is supported by a schema learning mechanism which constructs a hierarchy of competencies capable of dealing with problems of gradually increasing complexity. To allow for the learning of general concepts we introduce an algorithm for the generalisation of schemas from a small number of examples through parameterisation.

Throughout the robot's development its actions are driven by an intrinsic motivation system designed to mimic the play-like behaviour seen in infants. We suggest a possible approach to intrinsic motivation in a schema learning system and demonstrate how this can lead to the rapid unsupervised learning of both specific experiences and general concepts.

Contents

List of Figures

List of Tables

LIST OF TABLES

List of Algorithms

List of Equations

Chapter 1

Introduction

This thesis is concerned with the emergence of communication in artificial agents as an integrated part of a more general developmental progression.

Within this thesis we investigate two main aspects of communication, early gestural communication in the form of pointing and spoken language. In dealing with spoken language we focus on the association of words with both specific prior experiences and with more general concepts, drawing inspiration from the acquisition of language in infants. While we do not consider grammatical aspects of language learning here, we do believe that the schema learning mechanism employed within this work would lend itself well to this form of investigation in the future.

In describing communication as *"emerging"* we refer to the manner in which behaviours serving a communicative purpose develop out of a number of general purpose, non-communication specific faculties. For example in the early stages of development the combination of reaching behaviour directed towards distant objects whilst being in the presence of other social agents can result in the robot learning a pointing gesture.

We begin with our robots, shown in figure 1.1, having very little knowledge of their own bodies or of their environment and take them first through some low level motor learning, then on to gradually more complicated scenarios in which they can interact with objects and people.

However, we are not solely interested in just how communication can emerge, but also how this ability can feed back into more general learning. To this end we take the robot's development beyond the initial introduction of language into
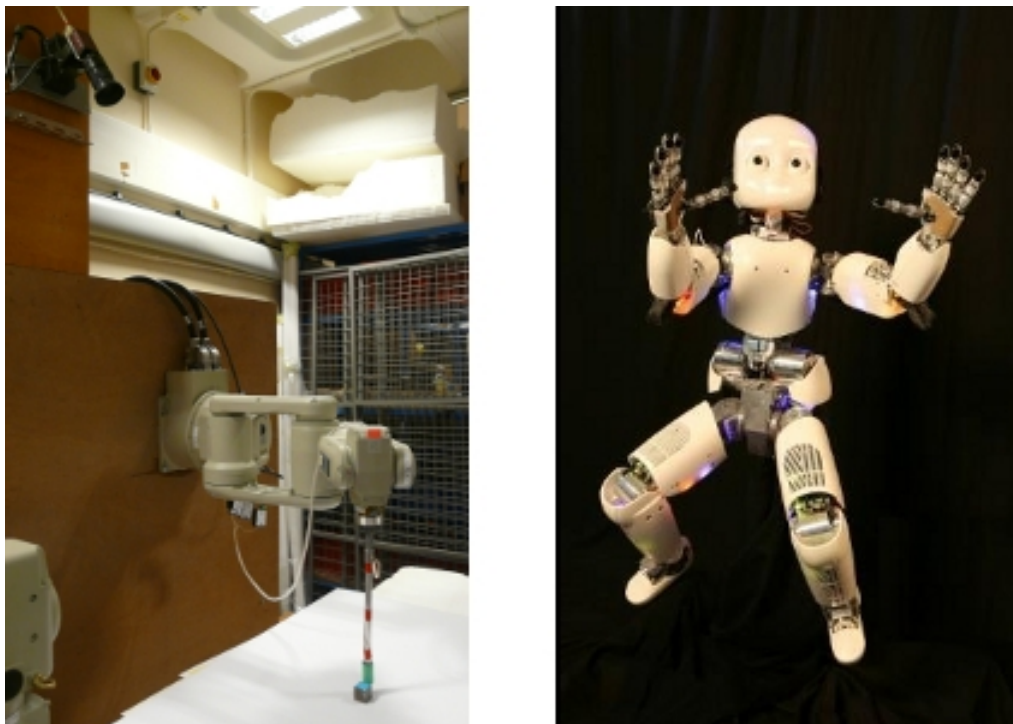
Figure 1.1: The two robot platforms upon which our experiments have been carried out. *Left:* An Adept six-axis arm with a single camera vision system. *Right:* An iCub humanoid robot with 53 degrees of freedom and stereo vision.

further motor learning, with the newly developed linguistic skills being used as a tool to help simplify these new and more complex scenarios.

An important consideration in a system attempting to partially mimic the learning stages of a child is how that agent's actions should be motivated. Play is an important part of any child's development, and we believe that it forms the basis of the learning mechanism used for training many of the basic competencies that we then rely on in later life.

While definitions of exactly what constitutes play behaviour vary, for our purposes we interpret play as being any intrinsically motivated behaviour. This is any behaviour that does not contribute to the satisfaction of some primary need or work toward some externally generated goal. The intrinsic motivation within our system primarily comes from an implicit goal of discovering further information about the world, resulting in our robots devising hypotheses about their experiences and then immediately being able to test them out in various different scenarios.

In this introductory chapter we outline the underlying theories that motivate our approach to this problem. We then discuss a number of key characteristics which we believe would be beneficial in any system attempting to investigate this area. The four main contributions made by this thesis are then briefly outlined. Finally we describe the general content to be found in the following chapters and the publications that have arisen out of the work undertaken as part of this thesis.

1.1 Motivation

We believe that for robots to be able to achieve a more advanced capacity for communication their entire concept of language must be firmly rooted in their sensorimotor experiences of the world. Rather than taking an existing robotic system and attempting to add the capacity for language on top we are more interested in having communicative abilities arise out of a more general learning system.

Instead of limiting our view of communication to linguistic acts we consider first simple forms of pointing which may emerge out of earlier play behaviour and form a stepping stone towards more expressive forms of communication.

To investigate this emergence of social understanding we create a system capable of experiencing a number of developmental stages. These stages are not limited only to those relating to communication as we believe that these earlier stages of learning play an important role in creating the mental competencies required for more social learning.

This social learning is then able to feed back into other aspects of the robot's learning. This allows for other agents (be they people, or other robots) to help scaffold the robot's mental processes through purely linguistic means, for example by guiding the robot's attention to a specific action or object at a crucial point in its exploration.

We believe that a system attempting to mimic the staged development seen in infants should also take advantage of the play driven learning behaviour that appears to serve human infants so well. To this end we have devised an approach to intrinsically motivated behaviour that allows the agent to generate actions relevant to the current environment without having any explicit goal, the play behaviour has the implicit goal of learning more about the environment instead of more traditional task specific goals generated extrinsically.

Children play throughout their development with ever increasing complexity, beginning with simple motor babbling and building up to complex play involving precise and skillful motor abilities, problem solving and social interaction. In a less ambitious but similar way our robot is able to make use of the early skills it develops to enhance the range of play behaviour available to it in later stages, thus building up a hierarchy of competencies capable of dealing with ever more complex environments.

1.2 Approach

In approaching this problem we have identified a number of characteristics that we believe would be particularly beneficial. These characteristics are outlined below and have informed the selection of techniques used to support our learning process.

1.2.1 Cumulative learning

Notable in human development is the way in which an infant's development may be classified into coarse stages, with each following stage building upon the competencies mastered in the previous one.

By its nature any system taking a staged developmental approach will need the capacity to learn cumulatively, as one of the key features of learning in developmental stages is that each successive stage builds upon the learning from earlier stages.

Throughout our experiments we expose our robots to problems of ever increasing complexity, using the knowledge gained in earlier scenarios these new scenarios can then be interpreted in terms of prior experiences. This greatly reduces the complexity of the learning problem and allows the robot to focus on the aspects of the environment that are novel.

1.2.2 On-line learning from sparse data

To be able to interact socially with other agents, especially humans, it is important that the robot be capable of learning quickly from a very small number of examples. A human teacher would typically lack the inclination to perform the same act hundreds or thousands of times, so learning algorithms that require a large number of samples are inappropriate for this domain.

Embodied agents have the ability to create the conditions necessary to test their own hypotheses. With this facility in mind we favour an approach to learning which results in the fast generation of general hypotheses from sparse data. We then allow the robot to conduct its own tests of these hypotheses through further action to determine their reliability.

1.2.3 Platform agnostic

The system should not be tied to a specific physical embodiment, as such it should have as little prior knowledge about its body as possible. Some amount of platform specific integration will be necessary to encode sensor information and possible actions, but the overall learning mechanism should not make any hardware specific assumptions.

To demonstrate this capacity within our system we have made use of two real robots, an Adept arm and camera system and an iCub humanoid robot, and two simulated robots which model the physical robots.

In each case the agent begins with very little knowledge about the world and learns how it can interact using the capabilities available to the physical embodiment it has been endowed with.

This goes beyond good engineering practices, also serving two valuable scientific purposes: it gives us the ability to place two different robotic systems in similar scenarios and discover how the representations arrived at by the agents differ; it also provides a check against the unintentional tailoring of our learning mechanism to a specific embodiment.

1.2.4 Self motivation

We believe that intrinsically motivated play behaviour is vital to learning in infants and that a similar approach can be beneficial to robotic agents. Children and young animals both engage in play-like behaviour, performing seemingly useless actions that do nothing to aid in the satisfaction of the more traditional primary needs (food, shelter, etc.) that are often the focus of extrinsically motivated learning mechanisms. However, while not directly contributing to the immediate satisfaction of any of these needs, play behaviour offers the agent, be they human, animal or robot, the ability to learn and practice new skills and competencies which may in the future prove to be very useful in achieving an extrinsically motivated goal.

A child learning to accurately hit the trunk of a tree when throwing stones may serve no direct purpose, but in the future when the child grows hungry that same skill may be employed to knock fruit from the branches of a tree that would otherwise be out of reach. It is not necessary that the child have this distant future goal of food collection in mind when first learning this skill, simply developing a new skill and becoming more proficient with it is suitably satisfying even in the absence of any more concrete goal.

In the absence of any explicit goals the robot should perform actions relevant both to the current environment and the robot's internal mental state, leading to the learning of novel information about the world. By allowing the robot to direct its own attention to the elements of the environment it finds most relevant

to its past experience it is able to perform its own experiments to determine the properties of the world and how objects and other agents within the world may be interacted with.

1.3 Contributions

The overarching theme of the research undertaken as part of this thesis has been how communication can first emerge and then influence developmental learning. A number of advances have been made to support this line of research, both in terms of the high level developmental stages experienced by our robots and in terms of advances to the learning and motivational mechanisms employed. The four key contributions made are introduced below and then discussed further in chapter 6.1.

1.3.1 Developmental progression

In this work we propose a developmental progression for robotic agents inspired by relevant psychological literature. The proposed progression focuses on a gradual transition from a 'newborn' state, in which the robot has little understanding of the world or of its own sensorimotor systems, towards the capacity for interacting with objects and understanding simple speech.

As part of this transition we investigate the possible emergence of pointing as an early communicative act resulting from prior learning experiences with the motor system.

We take a staged learning approach based around constraints, either in the form of reduced sensory input or simplified environments, which are lifted as the robot becomes habituated to its current stage of development.

In chapter 5.4 we show that the use of a staged developmental progression simplifies a number of problems allowing for expressive representations of both object and social interaction to emerge faster than would otherwise be possible with a similar learning mechanism.

At a high level the stages encountered by the robot can be summarised as:

- Motor babbling – In which the robot discovers the movements possible in its motor system.

- Motor vision mapping – The robot learns the visual result of the movements learnt in the prior stage.

- Object interaction – Objects are introduced for the robot to interact with through touching, grasping, moving and dropping.

- Proto-imperative pointing – In attempting to reach towards distant objects the robot discovers the act of pointing and finds that this can elicit assistance from other agents.

- Single word speech – Verbs and nouns are introduced separately.

- Multi-word speech – Verbs and nouns are combined to form simple sentences.

This progression is discussed in detail in chapter 2.

1.3.2 Algorithm for schema generalisation through parameterisation

To allow for the emergence of general schemas representing a class of possible actions we devised a generalisation mechanism that makes use of previous experience of similar actions to form hypotheses as to how these actions affect the world.

With a small number of examples the system is capable of forming generalised representations of its actions that allow it to predict the effects of an action in a novel scenario. This predictive capacity then allows the robot to plan more complex actions involving the use of multiple generalised schemas to achieve increasingly more advanced behaviours.

Chapter 3.6 covers the implementation of this algorithm, while 5.4 gives the results of a number of experiments which make use of this functionality and discusses the various implications of it.

1.3.3 Associated observations in schema learning

Causal schemas track information relating to the result of actions and the preconditions necessary for these actions to be successful. We have extended this

formalism to include the notion of associated observations which are events that may occur concurrently with an action but are neither a direct result of it nor a requirement for it to take place. This is particularly useful in the case of language which may relate to some aspect of the world or of the action being performed but is not itself directly responsible for a change in the environment.

Further discussion of this is found in chapter 3.2.4, while 5.9 shows the utility of this in the context of language learning.

1.3.4 Intrinsically motivated schema learning

The main driving force behind action within the framework is an intrinsic motivation system which excites schemas based on a combination of the novelty of the current experience and the similarity to past experiences. A particular sensorimotor memory is said to be excited if it partially or completely matches sensations currently being experienced. The level of excitation is determined by the novelty of the sensation combined with the similarity to a remembered sensation. The purpose of this excitation system is to steer the robot towards actions most likely to result in new learning experiences.

This mechanism provides the basis for our play behaviour — instead of acting towards some explicit external goal the robot's motivations steer it towards exploring the different ways in which it can interact with the environment. Combined with the cumulative learning approach this intrinsic motivation system often results in the selection of actions learned in earlier stages of development that have a high likelihood of being relevant to the aspects of the environment that had previously not been experienced. This results in the robot quickly learning how to represent more complex interactions which can then be used by the robot as a starting point for investigating the next stage of its development.

1.4 Thesis structure

A general guide to the content of each of the following chapters is outlined below:

1.4.1 Chapter 2: Developmental Learning

In this chapter we outline the developmental progression undertaken by our robots with reference to relevant developmental psychology literature and describe past robotic projects which have taken a similar approach. We also discuss the psychological inspiration for our intrinsically motivated play behaviour.

1.4.2 Chapter 3: Schema Learning

This chapter introduces the learning mechanism we have employed to support our investigation. We begin by discussing the prior use of schema models in the fields of psychology and computational modeling and then the introduction of methods for learning new schema models in robotic systems. We draw parallels between our schema system and aspects of the human brain. We then discuss the details of our implementation of the schema learning framework including the introduction of our generalisation mechanism, associated observations and an algorithm for intrinsic motivation based selection of schemas.

1.4.3 Chapter 4: Robotic Platforms

We describe our two robotic platforms, an Adept arm and camera system and an iCub humanoid robot, along with two simulated systems based upon these robots. We then describe the overall system architecture of which our schema learning system is a part.

1.4.4 Chapter 5: Experiments & Interpretation of Results

This chapter describes the experiments we performed upon the robots, which increase in complexity following the developmental progression outlined in chapter 2. We discuss the implications of these results and where possible we compare the differences in representation arrived at by the two different robots.

1.4.5 Chapter 6: Conclusions

In this chapter we summarise the contributions initially outlined in this introduction, drawing on illustrative examples from the experiments in chapter 5. Finally we discuss both a number of specific improvements that could be made to the

schema learning framework and some higher level suggestions of new investigations that could be made based around this work.

1.5   Publications

Parts of this thesis have been published in the following conference papers, copies of which can be found in appendix A:

- M. T. 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.

- M. T. Sheldon and M. Lee. PSchema: A developmental schema learning framework for embodied agents. In *Proceedings of the IEEE Joint International Conference on Development and Learning, and Epigenetic Robotics*, 7 pages, IEEE, 2011.

- M. Lee, J. Law, P. H. Shaw and M. T. Sheldon. An infant inspired model of reaching for a humanoid robot. In *Proceedings of the IEEE Joint International Conference on Development and Learning, and Epigenetic Robotics*, 6 pages, IEEE, 2012.

Additionally this work has been made use of in two European Commission Framework 7 projects, *Emergence of communication in Robots through Sensorimotor and Social Interaction (ROSSI)* and *Intrinsically Motivated Cumulative Learning Versatile Robots* (IM-CLeVeR). This usage is described further in the public deliverables of those two projects.

1. INTRODUCTION

Chapter 2

Developmental Learning

The developmental psychologist Piaget [42, 90, 91] categorised infant development into a series of stages, briefly summarised below:

1. Exercising sensorimotor schemata (*0 - 1 months*)

2. Primary circular reactions (*1 - 4 months*)

   - Child stumbles on an act that produces a new experience then repeats the act to reproduce the experience.

3. Secondary circular reactions (*4 - 8 months*)

   - Child performs actions associated with objects purely upon seeing the objects.

   - Actions are possibly representations of the object rather than actual attempts to act upon the object.

   - Concept of object permanence begins to develop in this stage.

   - Very beginnings of general space concept.

4. Co-ordination of secondary schemata (*8 - 12 months*)

   - Much clearer evidence of intention.

   - Larger separation between means and end. When completely separated means alone becomes play. When differentiated but related, problem-solving behaviour is exhibited.

- 'Actions' as representations from previous stage replaced by complex neural patterns. 'Motor meaning' has become 'symbolic meaning'.

- Object permanence is context bound.

- Object over-permanence - to the child the object still exists where it was first seen, despite having observed them being moved away.

- Cannot yet recognise what an object may look like from another point of view.

- Beginnings of understanding causality. Will wait for a parent to do things for it.

- Interest shifts from actions to effects.

5. Tertiary circular reactions (*12 - 18 months*)

- Proto-declarative pointing behaviour is established by this stage.

- Child engages in 'experiments' to discover new properties of objects and events.

- The 'spectacle' resulting from an action is now separate from the action itself.

- No longer exhibits over-permanence.

- 'True' imitation is exhibited.

- Actively solicits help from adults.

- Slowly ineffective actions drop away in a trial-and-error process.

6. Invention of new means through mental combination

- Child can follow displacements of objects even when the displacement takes place out of sight.

- Internal symbols make memory of past events, anticipation of future ones and reasoning about objects paths through space possible.

- Can infer cause from effect and vice-versa.

- Can imitate complex new models without extensive trial and error.

- Can imitate non-human and non-living objects.

- Can imitate absent objects.

When modelling infant development Piaget made extensive use of schemas, units of knowledge associating actions and perceptions. These structures, and their more recent use in the field of artificial intelligence, are discussed in more detail in chapter 3.

The progression through the Piagetian stages can perhaps also be linked to Vygotsky's *'zone of proximal development'* [111], although this concept is usually applied to later development. The zone of proximal development being the difference between the already completed level of development and the potential for achieving elements of the next developmental level given appropriate guidance or *'scaffolding'* [112].

Erez and Smart [32] describe a number of different approaches to *'shaping'* or scaffolding of an agent's learning process. Both in terms of modifying internal factors of the agent and modification of the environment in which the agent interacts. Our system first undergoes learning constrained by internal factors as it learns how its body functions. After this we scaffold the robot's environment, first introducing objects one at a time and then introducing spoken language alongside the robot's interactions. Finally we make use of the language understanding the robot has developed to scaffold its behaviour through spoken guidance, while the robot learns to form a stack of multiple objects.

Pardowitz and Dillmann [86] discuss an alternative Piagetian approach to life-long learning based around a *'programming by demonstration'* mechanism. They introduce methods for determining the similarity of tasks, allowing the system to make use of knowledge gained from completing related tasks to assist with new ones. However a full developmental approach is not taken, with the system starting with predefined high level knowledge about object relationships and object classes.

Law, et al. [60] provide a detailed review of infant development summarised into a timeline beginning with prenatal development and continuing through postnatal development with a view towards robotic implementations. While our own work focuses on a postnatal period of development with certain capabilities such as arm movements and hand motions already available to the robot, these need not necessarily be considered as innate knowledge. Kuniyoshi and Sangawa [59] show that basic motor patterns can emerge during prenatal development, this is

demonstrated through the simulation of a fetal model consisting of a musculo-skeletal system with a basic nervous system suspended in a uterine environment. So while we do provide a small number of basic motor primitives to our robots, we do not argue that these are necessarily innate in children, and that rather they may emerge during an earlier stage of prenatal development which is not modelled within our own experiments.

Guerin [46] provides a survey of developmentally inspired approaches to artificial intelligence, although this focuses primarily on very early infant learning, and excludes language learning. An earlier survey by Lungarella, et al. [67] does include developmentally inspired language learning, but the majority of these studies begin at a much later stage of development than is introduced here, with many underlying competencies preprogrammed into the agents. In this work we attempt to demonstrate how the learning of communication can be interwoven with early motor learning. The sensorimotor experiences encountered in the earlier stages provide a grounding for the linguistic aspects, which in turn can be used for guiding the agent's attention in further motor learning tasks.

Asada, et al. [4] survey the state of cognitive developmental robotics [5], which has a strong focus on the development of neural representations through interaction between the physical embodiment of the robot and the environment, progressing from prenatal development onwards and continuing in to social development. Unlike the earlier work surveyed by Lungarella, et al. this approach attempts to bridge the gap between early developmental motor learning and later social learning. Our work also attempts to bridge this gap, investigating how the learning of early motor skills can influence communication in the form of pointing and how later language learning can add another means by which a caregiver can scaffold physical interactions, however we take a more symbolic approach drawing on relevant psychological literature for inspiration.

2.1 Intrinsic motivation

When acting based upon intrinsic motivation, an agent performs actions motivated by an internal sense of interest in that action itself, as opposed to acting in an attempt to achieve some separate goal, as in the case of extrinsic motiva-

tion [96]. Of particular interest to us is the intrinsic motivation of play behaviour seen throughout a person's life, but with special emphasis on infant play [61].

Oudeyer, et al. [81, 82, 83] summarise intrinsic motivation from both a psychological and robotic perspective, and attempt to classify the different approaches taken. They also explore the intrinsic motivation of language learning rooted in play and curiosity [80], using a framework based around 'progress niches', which are similar to Vygotsky's concept of the zone of proximal development. They show how an intrinsic motivation system can allow a robot to self-organise its learning process.

The intrinsic motivation system which drives action within the work presented here is inspired by play behaviour seen in infants, which we view as playing an important role in motivating learning. Rather than having explicit goals that the robot tries to achieve, the motivation system makes the execution of actions relevant to novel experiences interesting in an attempt to learn new ways of interacting with the world. The motivation system attempts to select the schema which has the most in common with the novel aspects of the environment currently being experienced, in effect being reminded of actions that it had previously used in a similar, but not identical, context. For example, when touching an object for the first time the sensation may remind the robot of the feeling of touching its own hand when performing grasping motions without objects present. The triggering of the action associated with this memory then leads to the robot experiencing new sensations related to holding an object. An additional schema is then formed to represent this new experience. Implementation details of this system can be found in chapter 3.4.

This approach differs from the use of intrinsic motivation within the reinforcement learning community [8, 99, 103] in which an internally generated signal rewards a recent action and so motivates its repetition for further learning. By contrast our motivation happens prior to an action taking place, the robot is motivated to perform actions which remind it of some aspects of the current environment, previously performed actions have no effect on this motivation value other than through the changes they make in the environment and through the decreased novelty of the actions in circumstances that do not elicit further learning.

Our implementation of intrinsically motivated play behaviour can also be viewed in terms of Piagetian *'assimilation'* and *'accommodation'*. Assimilation is the process by which a child makes use of existing knowledge to interact with the environment, while accommodation extends or modifies this knowledge to represent previously unknown properties of the interaction. For example, a child being introduced to a rattle may utilise its previous experience of grasping various different objects to grip the rattle, this would constitute assimilation of the action of grasping the rattle by its existing mental structures. During this interaction the rattle might make unexpected sounds as the child moves it around, resulting in accommodation taking place as the child learns this new effect.

2.2 Lift Constraint, Act, Saturate

The system implements a Lift Constraint, Act, Saturate (LCAS) [63, 64] 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 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 affect its senses, an approach similar to the scaffolding performed by parents when helping children to learn.

By allowing the agent to habituate between developmental stages, the system is given 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.

2.3 Developmental progression

The following six stages outline a possible robotic developmental progression leading from a 'new born' state to simple linguistic communication.

2.3.1 Motor babbling

In early infancy children produce and repeat rhythmic motions of their limbs [55], conceptually similar to the repetitive vocal babbling that infants display when learning to produce more meaningful linguistic sounds [78] and potentially sharing the same underlying mechanism, with manual babbling possibly assisting in the learning of vocal babbling [31, 66]. Iverson and Fagan [53] go on to suggest that this relationship between manual and vocal babbling may form the basis for the generation of spontaneous hand gestures alongside speech which carry on into adulthood.

In this initial stage the robot has had no prior experience of the world or of its own body. It performs spontaneous motor actions in order to discover the properties of its motor systems and its anatomical constraints.

The form the motor babbling takes varies between the two robotic platforms. The iCub begins by first learning eye and then head saccades, allowing it to develop a gaze space based upon the combined head and eye positions. After this it also performs reaching, pressing, grasping and releasing motions in the absence of any objects. On the Adept robot this consists of making movements of the arm and learning the different possible joint configurations.

2.3.2 Motor vision mapping

Both robots then perform the previously learnt arm movements whilst fixating upon either the hand in the case of the iCub, or the finger in the case of the Adept. This gives the robots the opportunity to learn a mapping between the movements made and the changes this creates in the robot's vision system.

This allows the robots to reach out to touch (or point towards) objects which are detected visually in the locations in which it has previously experienced its own hand or finger. This results in visually elicited, but not visually guided [21], reaching as a consequence of the intrinsic motivation system being excited by the object's presence in a location that the hand has previously reached towards.

Further details on the motor babbling and vision mapping stages undergone by the different robots can be found in chapter 4.

2.3.3 Object interaction

Once the robots have learnt the results of moving their end effectors into different locations we introduce objects for them to interact with.

The Adept arm is only capable of reaching out to touch objects, however the iCub also has the capacity to grasp, move, drop and press objects.

When first touching or grasping objects the robot receives new, unexpected, sensations from its touch sensors. This leads to the formation of new schemas representing this knowledge. Because of the way in which the intrinsic motivation system favours both novel sensations and novel actions this leads to the robot repeating these new interactions a number of times. Similar behaviour can be seen in infants in what Piaget terms *'primary circular reactions'*.

Interest in performing these repeated actions will gradually diminish as they become less novel, at which point the robot will begin exercising other schemas. If the object is removed and the robot continues exercising its existing schemas, the excitation produced by these schemas will also be reduced further. When the object is then reintroduced to the robot it is exciting enough, in comparison to other available actions, to draw the robot's attention back to executing schemas related to the object. This results in the robot exhibiting a behaviour that mimics some aspects of the *'secondary circular reactions'* in infants.

2.3.4 Failed grasping leading to pointing

Vygotsky suggested that pointing develops out of a failed grasping behaviour in which the child attempts to reach for an object which is too far away and the parent interprets this as the child pointing at a desired object and as such fetches the object for the child, thus associating a new meaning with the act of reaching for a distant object [65, 111]. Initially all social meaning of this act is inferred entirely by the parent, the infant is making a real attempt to reach the object and failing, but through the actions of the parent the infant comes to associate the same communicative meaning.

This has been classified by many researchers as *proto-imperative pointing* or *ritualised grasping*, used by the child to indicate an object of desire to a nearby adult. By around 12 months the child has also learnt to perform *proto-declarative pointing* which is used to acquire joint attention on an object with an adult. Masataka [70] provides evidence to indicate that proto-imperative pointing and proto-declarative pointing may follow different developmental paths, with proto-declarative pointing emerging out of the earlier index finger extension behaviour that infants exhibit when exploring reachable objects. All pointing within this study is modelled on proto-imperative pointing, with the robot using this pointing gesture to request distant objects.

Povinelli, et al. [92] show that while chimpanzees raised in captivity can be trained to perform proto-imperative pointing they do not appear to make the jump to proto-declarative pointing.

Tomasello, et al. [109] argue that infants may possess a much deeper social understanding at this stage than previously thought, able to communicate a great deal through pre-linguistic gestures such as pointing.

In attempting to touch objects that lie outside of the robot's work-envelope, it incidentally performs what looks, to a human observer, like a pointing motion. Through assistance from a human observer, fetching the indicated object for the robot, the robot's representation of this action moves away from being a direct attempt at manipulating the world towards an attempt at social communication.

2.3.5   Language learning

Arbib [2] suggests a schema based model for the acquisition of language, discussing how a schema based interpretation can remove the need for an innate universal grammar as proposed by Chomsky [20]. In this thesis we consider only the association of words with sensorimotor experiences, but future work could extend this to consider grammatical implications.

Iverson and Goldin-Meadow [54] describe the early developmental path of infants learning to communicate verbally. They show that in most cases infants follow a consistent progression from pointing to two word speech.

Dautenhahn and Billard [26] contrast robotic approaches based around either Piagetian or Vygotskian ideas of social development. While our general approach to infant development is highly influenced by Piagetian stages of development

and schema based modelling, our view of social situated development is more Vygotskian, in that we consider it to be of great importance to the emergence of higher cognitive functions. As such we attempt to interweave general motor learning and the learning of communication within this work, with each providing influence upon the other.

Steels, et al. [106] show that the concept formation process of agents must be based on similar sensor input and result in similar conceptual repertoires for communication to develop in a population of agents. It also shows that once a lexical system is in place it can overcome the randomness inherent in verbal communication. A population of thousands of agents is used, although only two agents are embodied at any one time. The large population allows for the occurrence polysemy and gives the ability to study the resolution of the resulting multiple meanings within the population.

In addition to this Steels has investigated evolutionary mechanisms for the emergence of language within a population [105], looking at the possibilities of horizontal and vertical transmission of language within a culture of agents and the ways in which *'language games'* (patterns of joint linguistic behaviour) become ritualised.

Oudeyer and Delaunay [79] extended Steels' language experiment to add a developmental learning aspect which regulates the growth of each agent's lexicon depending on their prior success with related meanings. This results in considerably faster convergence upon a shared lexicon within the agent population.

Tikhanoff, et al. [108] demonstrate the learning of language on a simulated iCub, after also learning reaching and grasping behaviours through the training of neural networks. However the action selection performed is limited to a small number of predefined sequences, either reaching, grasping or dropping an object. In contrast our own approach, whilst also limited to similar basic actions, is able to combine these in novel ways and learn the result of these new composite interactions.

Fong, et al. [35] provide a survey of social robots and learning techniques including scaffolding, direct tutelage, imitation and goal emulation. Within our own experiments we focus primarily upon scaffolding, first of the environment that the robot is exposed to and then later through linguistic means, offering guidance

on actions which may provide further stimulation in more complex environments through the use of language learnt during earlier stages of development.

Harnad [50] discusses the symbol grounding problem in the learning of language. This covers the problem of how semantic interpretation can be made intrinsic to the system, not just parasitic on the meanings in the minds of observers and suggests that symbols must be grounded in a behavioural system that can describe the objects and states of affairs that a symbol refers to. We attempt to achieve this within our own work by integrating the learning of communicative abilities, both gestural and linguistic, within our main sensorimotor learning mechanisms. Making it possible for the meaning of words to emerge out of related sensorimotor experiences.

Zlatev [115] explores the philosophical implications of the symbol grounding problem and Searle's argument that artificial systems do not posses any form of communicative intention [100]. Zlatev concludes that by taking a Vygotskian view that *"intentionality, self-consciousness and meaning are real emergent properties arising from the dialectical interaction between specific biological structures (embodiment) and culture (situatedness) through a specific history of development (epigenesis)"*, then an embodied, socially situated, developmental robot can potentially be said to have intention.

Within our progression language learning is divided into two distinct stages, outlined below.

2.3.5.1 Single word speech

The robot is provided with auditory input (reduced to a text token by speech recognition software) while it acts upon different objects presented to it. The words heard during these interaction may relate either to the object being acted upon (such as that object's name, or some properties of the object) or to the action being performed.

During early language development infants acquire the ability to learn new words from a very small number of examples, often referred to as *'fast-mapping'* [17, 71], the meaning of these words can then be refined over subsequent exposure to them in varying contexts. Yu and Smith demonstrate that both adults [113] and 12 month old infants [104] can make use of multiple encounters with words from varying situations to help disambiguate their potential meanings. Yu, et al. [114]

then go on to develop and compare a number of computational models, through which they suggest that there may be a common mechanism supporting both the hypothesis testing theory of language acquisition, in which the agent forms hypotheses in ambiguous circumstances which can later be tested, and associative learning, in which statistics are built up over a number of experiences.

Within this work we implement an approach to learning based around the formation of hypotheses from sparse data, used in both motor and language learning, which the robot is then able to test and refine through further interaction. However the viability of these hypotheses is determined based on the statistical information accumulated during interactions with the environment and other agents. This provides our robot with a similar capacity for the fast learning of language from interaction with human social partners, with these early mappings then being refined based upon further experience.

2.3.5.2 Multi-word sentences

Finally the learnt speech can be combined to allow the robot to react to simple sentences. The combination of verbs and nouns or adjectives allows for more refined selection of generalised schemas, specifying which possible interpretation of a generalised action should be executed.

For example, hearing the words *'grasp red'* causes the robot to select a previously learnt generalised grasping schema. This schema has the potential to be instantiated in a number of different ways depending on the environment it is being executed in. If multiple objects are present then the generalised grasping schema could be used to grasp any of them, with the addition of the word *'red'*, the instantiation of the grasping schema which includes interaction with the red object becomes more exciting than other potential interpretations.

Chapter 3

Schema Learning

In the broadest sense a schema is a grouping of information relating to a concept, object or event which may be organised hierarchically. Structure of this sort was first introduced by Kant [56] and has since seen considerable use in the field of psychology as a tool for formalising models of behaviour and learning.

Bartlett played a large role in bringing the type of schema based reasoning that is more familiar today into modern psychology with his investigation into memory storage and recall in which he argued for the interpretation of memories and their reconstruction based around a filter of previously learnt schemas [7].

In this work we deal primarily with causal schemas, relating to events. These 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 sensor values specified in that schema's prediction.

In its simplest form a causal schema consists of a set of pre-conditions, an action and a set of post-conditions, providing a basic forward learning model. An example of this schema interpretation can be seen in figure 3.1.

Pre-conditions	Action	Post-conditions
Object on table	Reach to object	Object on table
		Touching object

Figure 3.1: A high level example of a simple schema, as used in our learning framework.

3. SCHEMA LEARNING

Often in psychological and cognitive modelling, schema populations are constructed manually and begin at higher levels of abstraction than we would typically see in a robotic system. In this work we concern ourselves with the automatic learning of schemas, beginning with very low level motor movements and building up towards more general high level concepts.

Arbib builds his initial computational schema interpretation based upon output feature clusters [1], which encode the features relevant for specific interactions between an agent and an object. Later he expands on this [2] and discusses how schemas can form a useful bridge between lower level computational implementations, in his example neural networks, and higher level psychological analysis.

An early computational approach to the learning of schemas was introduced by Becker [9] in his JCM architecture. In contrast to Drescher's [30] later formalism of pre-condition, action, post-condition triplets with sensory information consisting of vectors of binary sensors, the kernels which Becker's schemas were constructed from encapsulated either sensory information or actions in any order. While they can be ordered to mimic Drescher's formalism, as shown in figure 3.2, they can also be ordered to contain multiple intermixed actions and sensations, as in the example in figure 3.3.

$$<\text{SensoryKernel}> \rightarrow <\text{ActionKernel}> \Rightarrow <\text{ResultingSensoryKernel}>$$

Figure 3.2: An example of a Becker-style schema which matches the pre-condition, action, post-condition definition of a schema.

$$<\text{LiftBall}> \rightarrow <\text{BallIsLight}> \rightarrow <\text{ThrowBall}> \Rightarrow <\text{BallIsFarAway}>$$

Figure 3.3: An example of a Becker-style schema in which an action is used to expose further sensory information prior to another action being taken which provides the final result.

Drescher's schema mechanism introduced the concept of 'synthetic items' which were schemas that could be used as pre-conditions within other schemas. This gave his system the capacity to test for properties of the environment which were not immediately accessible, whilst still preserving the pre-condition, action, post-condition structure.

Using Becker's model as a starting point, Bond and Mott [13] produced the first example of schema learning within a robotic context, using schema learning

to enable a mobile robot to learn simple navigation activities aimed at alleviating the robot's 'hunger' by reaching a charging station.

A common criticism of Drescher's approach is that it fails to scale to large numbers of schemas due to the requirement of tracking and updating large amounts of potentially unrelated sensor data for each schema. Foner and Maes [34] address this by introducing some global heuristics which can be used to limit the amount of information that requires updating. We take a similar approach limiting the updating of schemas to only the currently active schema, however our excitation mechanism is tested against all schemas, reintroducing this issue elsewhere. Rather than limit the schemas tested for excitation we instead introduce a generalisation mechanism [102] which allows the system to create more expressive hypotheses about the environment and its actions within it and so require far fewer schemas to achieve this representation.

This approach to generalisation allows the system to build more complicated skills based upon the execution of existing simpler schemas. Although based around a neural network based approach, Ring's [94] CHILD system introduces a conceptually similar approach. By introducing the robot to problems of gradually increasing complexity and allowing the knowledge gained from earlier learning to influence the robot's action we are able to incrementally construct a hierarchy of skills.

Chaput, et al. [18, 19] created a neural based implementation of Drescher's original schema learning mechanism making use of self-organising maps. However this approach still has the original limitation of only supporting binary sensors within the system.

Holmes and Isbell [51] extended Drescher's work to enable the use of continuous value sensors. They showed that it was possible to model Partially Observable Markov Decision Processes (POMDPs) via this mechanism.

Rather than use the binary sensor mechanism of Drescher or the continuous value sensors that Holmes and Isbell employed our system makes use of discrete sensor values, using a field based sensorimotor system which divides the sensor and motor spaces into potentially connected fields.

Cooper and Shallice [23] produced a computation implementation of a schema based theory of action selection, with which they performed a number of computational modelling tasks. These tasks were based around a high-level scenario

involving the use of actions and objects specifically relating to the task of preparing coffee. Cooper and Shallice [24] also demonstrate the continued utility of explicit schema structures when compared to attempts to arrive at similar emergent properties in artificial neural networks.

Cooper and Glasspool [22] used this framework in an early investigation into the selection of schemas based upon the potential actions afforded by the environment and discuss possible future enhancements to mimic child-like learning more closely by employing a notion of 'boredom' to direct the agent towards more novel stimuli. With the proposed enhancements this approach may be similar at a high level to the intrinsically motivated play behaviour that drives our own system.

Guerin [47, 48] has since used a schema based approach in a simple simulated robotic environment, learning grasping tasks. He introduces 'super-schemas' as a mechanism by which multiple schemas can be combined to avoid unnecessary additional learning of related actions, for example moving a hand whilst also gripping an object. This has some similarity with the use of target actions within our generalisation mechanism, which allow the robot to make use of existing schemas to assist in the execution of new, more general schemas.

Perotto, et al. [87, 88, 89] introduce a Constructivist Anticipatory Learning Mechanism (CALM), which makes use of a schema learning approach. 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. As with Drescher's schema mechanism this is limited to binary sensor values.

Object Action Complexes (OACs) [38, 58] provide a similar formalism to schemas, mapping actions performed upon objects to their effects, with the state of the world resulting from the execution of an OACs kernel being stored in terms of STRIPS rules [33, 74]. Although the OACs formalism itself allows for many different implementation approaches, the method taken by Krueger, et al. [58] requires the training of prediction functions using hundreds or thousands of trials by the robot to converge on a low error rate. By contrast our approach, based around staged learning with hypothesis generation from small numbers of examples, allows for the learning of general action effects from very few trials. We believe this makes our approach more suitable for social learning, as human

social partners are only likely to repeat the same interaction a small number of times.

Sahin, et al. [97] utilise another schema-like structure as part of an investigation in to affordances. An affordance represents the possible interactions presented to a specific agent by properties of the environment [41], for example to a human a small box may afford the interaction of lifting, however to a mouse the same box may afford the interaction of climbing inside. Figure 3.4 shows the general form of an affordance as modelled by Sahin, et al. When an agent performs the specified behaviour upon an entity it expects to encounter the predicted effect.

(effect, (entity, behaviour))

Figure 3.4: A general representation of an affordance as formalised by Sahin, et al.

(lifted, (red-ball, lift-with-left-hand))

Figure 3.5: The affordance of lifting presented by a red ball.

Figure 3.5 shows a specific example of this form of affordance, in which a red ball, when picked up will exhibit the effect of being lifted. Figure 3.6 then shows a version of this affordance in which the colour component is discarded, allowing the affordance to represent the lifting of any colour of ball. This has similarities with our own generalisation process, however while both allow for the agent to ignore properties which do not impact upon the action being undertaken, our own parameterisation process allows more refined relationships to be discovered between properties of the environment and the action undertaken. For example, when learning to touch objects our system is able to arrive at a general schema which states that if an object is observed in a location $x, $y and the robot performs an action resulting in its hand also being observed in that same location then it will receive a touching sensation from coming in to contact with the object. In this way the $x, $y parameters observed from the object's visual properties are used as a component of the action. The process by which these generalised schemas are formed and utilised is discussed further in section 3.6.

(lifted, (< *-ball>, lift-with-left-hand))

Figure 3.6: An affordance repesenting the lifiting of any colour of ball.

Georgeon, et al. [39, 40] suggest an approach to learning involving intrinsic motivation within a schema learning system. The schema definition employed by Georgeon and Ritter differs from that used in the majority of Drescher influenced schema systems, rather than consisting of a set of pre-conditions, an action and a set of post-conditions, their schemas are recorded in terms of pairs of interactions, meaning that if the first interaction has occurred, then the second interaction can be enacted. The intrinsic motivation aspect of this work is inspired by the philosophical concept of 'inversion of reasoning' [27, 28]. Rather than be rewarded for achieving a high level goal, the simplest interactions within the framework are given predefined 'proclivities', for example moving forwards may have a positive proclivity value, whereas encountering a wall may have a negative value. The motivation for executing a high level schema, consisting of multiple interactions, is then arrived at by the combined value of the low level proclivities. The approach taken within our own system takes a different view of intrinsic motivation, combining the novelty of current experiences with their similarity to past experience, resulting in a selection mechanism which we believe aids the agent in selecting actions likely to elicit further novel experiences.

3.1 Cognitive model
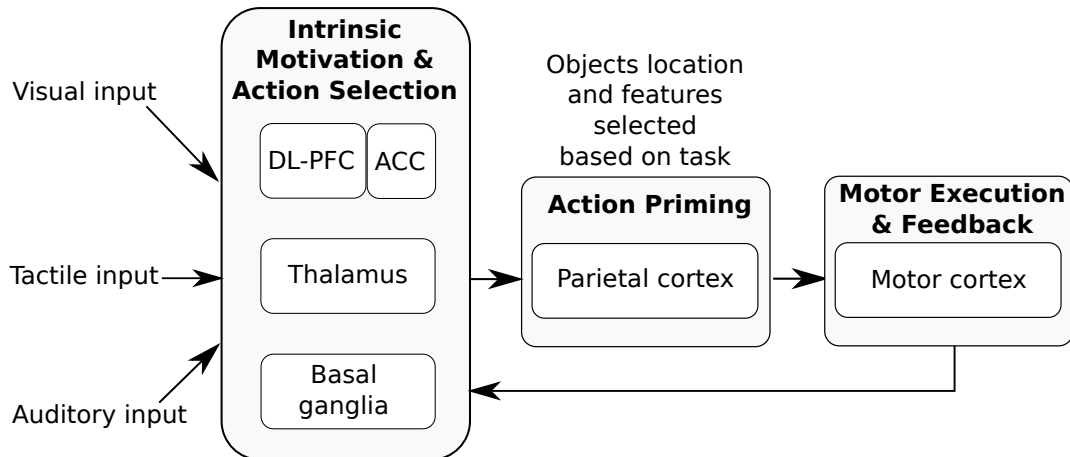
Figure 3.7: Brain regions and their interactions relevant to motivation, action planning and action execution. *DL-PFC* - Dorsolateral prefrontal cortex. *ACC* - Anterior cingulate cortex.

When designing our schema system we have tried to draw inspiration from a neuropsychological view of the human brain, a high level overview of the brain regions and interactions that we base our system upon can be seen in figure 3.7.

This model focuses on brain regions relating to three main areas of interest in our work: Motivation, action planning and action execution. The role of these areas within that context is discussed below.

3.1.1 Basal ganglia

The basal ganglia play a role in both low level motor control and higher level cognitive processes, being both connected with the motor system and connecting via the thalamus to the cerebral cortex [73].

Redgrave, et al. [93] account for the basal ganglia's activation in a wide range of different scenarios by suggesting that they function as an action selection mechanism, mediating the access to both motor and cognitive systems between multiple competing drives. They propose a winner-takes-all resolution to the conflicting demands placed upon these systems, with this action selection taking place at different levels of decision making. A similar approach is employed by our schema framework, on one level the excitation system causes selection based on

the saliency of stimuli when considered from the perspective of knowledge-seeking play behaviour. While many schemas may be excited by the current stimuli only the most salient one is selected for execution, the salience of the action and the stimuli involved in the action are then decreased as they become less novel allowing for the switching to the previously less exciting competing schemas. On a higher level the chaining facility causes selection to be driven by a combination of a schema's ability to contribute towards the current goal and its reliability judged by past experience.

In addition to this Graybiel [43] discusses the basal ganglia's role in 'action chunking', forming higher level sensorimotor units out of low level information, which can be seen as being in some ways analogous to our schema creation.

3.1.2   Dorsolateral prefrontal cortex

The dorsolateral prefrontal cortex (DL-PFC) plays a role in working memory [36], attention and planning.

It is comparable to a number of aspects of our system, the novelty based saliency mechanism which drives our intrinsic motivation, the schema chaining functionality and the association of groups of observations.

Blumenfeld and Ranganath [10, 11] show how the DL-PFC may assist in the association of information by keeping them bound together prior to being recorded in long term memory. This is separate from the binding of coincidental sensations that occurs in the hippocampus which allows for the recollection of a specific event (episodic memory) and instead allows for encoding and retrieval of sensory information relevant to the current action or task [6].

Buckner, et al. [15] demonstrate that there is decreased activation in this region for previously observed sensations (in their case specifically targeting verbal input). The schema system also has a different response to new groups of observations and to previously experienced groupings. With new groupings new schemas (or new 'memories') are formed, whereas when groups of observations that have already been encountered are experienced again they instead trigger a less exciting recollection of the previous actions that occurred alongside them.

Dagher, et al. [25] show that the DL-PFC plays an important role in the sequencing of behaviours at a higher cognitive level than just motor planning, finding that the level of activation in these region correlated with the complexity

of the task being attempted. This has similarities with the schema chaining functionality within our system, which allows it to connect together multiple schemas to achieve a goal state. The previously mentioned capacity for considering task specific information also comes into play when forming schema chains, as the system must consider what information is relevant in each stage of its plan for reaching the next stage in the chain.

MacDonald, et al. [68] reinforce the hypothesis that the anterior cingulate cortex (ACC) monitors performance for inconsistencies and undesirable outcomes and in such cases signals the DL-PFC, causing more high level planning to take place. When the expected outcomes of an action fail to occur in our system this triggers a re-evaluation of the schema chain currently being executed, allowing it to reconsider the best course of action in a similar manner.

3.1.3 Thalamus

The Thalamus receives sensory information, reorganises it and then relays it to different brain regions. Through cortical loops involving the basal ganglia and DL-PFC it plays a role in many of the activities previously discussed in those regions. The schema system separates out information based on the sensory channel it was received via, allowing for both domain specific and multimodal comparisons or generalisations.

3.1.4 Parietal areas

A large role of the parietal regions is the integration of information across multiple different sensor modalities.

Grefkes, et al. show that the medial intraparietal cortex performs translations between visual and motor spaces [44], and similarly the posterior parietal cortex has previously been found to provide this functionality in macaques [16]. This capacity for translating between the gaze space and the motor space is similar to the motor-vision mappings which the system learns. With the Adept robot these mappings are learnt within the schema memory, whereas with the iCub a more comprehensive mapping process takes place more closely modelled upon the behaviour of infants. Both of these approaches are described in chapter 4.

The anterior intraparietal region in monkeys shows considerable activation during tasks relating to the extraction of feature information about objects for the purposes of grasping [77], with the human equivalent showing similar properties [45].

By extracting an object's locational and feature information from the current perception of the world, generalised schemas can be resolved to specific motor actions tailored to the current task immediately prior to their execution by the motor system.

3.1.5 The mirror neuron system

The mirror neuron system was first discovered in the brains of monkeys [37, 95] and later studies showed a similar system at work in the human brain [75]. A mirror neuron is a neuron which fires both upon the execution of an action and upon the observation of another agent performing a similar action. Each mirror neuron is paired with a canonical neuron, however the canonical neuron is only activated during the execution of an action and not during its observation. This has prompted speculation that the mirror neuron system may have been crucial in the evolution of language [3].

Tettamanti, et al. [107] show that listening to action related sentences can trigger a mirror neuron response in humans and Kohler, et al. [57] have previously found that a noise related to an action can trigger a response in monkeys. This adds further weight to the idea that the mirror neuron system encodes action content at an abstract level and that this content can be activated auditorily. This suggests that language is strongly linked to the sensorimotor system. While it is possible that subjects were just engaged in motor imagery this is partially ruled out by the lack of other brain activity which normally accompanies these processes.

A study by Buccino, et al. [14] suggests that mirror neuron responses only occur for actions that the observer can duplicate. For example, humans watching a dog biting will show frontal parietal activity, while they will not when watching a dog bark. This also shows that the mirror neuron system generalises to different species, possibly suggesting that the goal of the action has a much greater effect than the observation of the action itself.

The goal directed nature of mirror neurons is further reinforced by a study by Umiltà, et al. [110] in which the neural response from monkeys was measured when they observed the experimenter grasping an object and when they mimed the grasps with no object present. It was found that the mimed grasp produced no response, while the real grasp did. It was also found that if the view of the object was occluded so the final stage of the grasp wasn't visible then some response was still produced, suggesting the goal was being inferred from the action.

Oztop and Arbib [84] hypothesise that the mirror neuron system may have evolved to provide feedback for visually directed grasping with the social usage being an exaptation[1] occurring when this became applied to the hands of others.

Oztop, Kawato and Arbib [85] provide a computationally guided review of mirror neuron literature and provide box diagrams of a computational model called the MNS model. Bonaiuto, et al. [12] have made attempts to extend this model, creating a more comprehensive version titled MNS2.

Murata, et al. [76] show that the canonical neurons, in addition to encoding self-action (as opposed to observed action, encoded by the mirror neurons), also encode object representations in monkeys. It has been hypothesised that canonical neurons may play a large part in understanding nouns while mirror neurons are used in the representation of verbs [69].

We encode all language related information within the schema framework, tying it directly to the sensorimotor experiences it represents and throughout the developmental progression we attempt to weave the learning of communication and general motor skills together. The schema framework also has the capacity to recognise the actions of others and relate them to the robot's own experiences based around the shared understanding of the goal driving the action.

3.2   Association of sensations and actions

The schema memory associates actions with the sensations that appear to result from that action and the sensations which trigger the action. In addition to this, sensations may become associated with other sensations that have been seen at the same time (associated observations). This has the effect that one set of

---

[1]An exaptation being the exploitation of an evolutionary adaptation to serve a different purpose than the one it initially developed for.

sensory stimulation may remind the agent of a past action it performed and so cause the agent to perform it again.

The direct associations between actions and sensations provide the mechanism by which the schema memory is able to predict the outcomes of its actions and form plans to achieve changes in the environment, while the associated observations can be used for the learning of language with verbs becoming associated with actions, nouns becoming associated with groups of sensations and adjectives becoming associated with individual sensations. Novel combinations of words can lead to the agent not only performing a previously remembered action, but also to performing new actions based on the combination of memories stimulated by the different words.

3.2.1 Observations and world states

Rather than attempting to directly encode sensor information within a schema we combine related information together into *'observations'*. Each observation has a type and a number of properties, these properties being defined by the type of observation. For example, a visual observation may include the spatial position of an object, the object's colour and an object identifier passed to the system by a lower level vision processing unit, while an auditory observation may simply have a single property containing whichever word the robot has recently heard, with multiple auditory observations occurring in the case of a sentence being heard.

These observations are collected together in a *'world state'*. A world state being little more than a set of observations describing the state of the environment as experienced by the robot at a given point in time.

3.2.2 Observation probability tracking

In addition to tracking the probability of a schema's success as a whole, the schema memory 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 and predict the most likely of the two each time the schema is evaluated.

3.2.3 Observation disappearance

In the case where an object or sensation disappears, or fails to reappear when expected, the underlying sensory system is required to report this disappearance explicitly to the schema memory. It is not enough that this sensation simply no longer be reported as the schema memory only creates new schema representations based on new or altered observations, not on the absence of previously experienced ones.

This has the advantage that the lower level short-term memory is then able to 'forget' remembered sensations if they are not experienced for a short while. For example, if an object goes out of view during the early stages of development, prior to any strong understanding of object permanence, it may simply be forgotten about without creating unnecessary, and arguably incorrect, schemas implying that a head or eye movement can cause an object to be removed from the world. As the understanding of object permanence increases the object might still be reported to the schema memory for gradually lengthening amounts of time since it was last seen, but eventually it may still be forgotten by the short-term memory. This further separates the action which may have initially caused the object to leave the view from the eventual absence of knowledge about it caused by the short-term memory forgetting it.

3.2.4 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 5.9.

There are two different mechanisms by which an observation may become associated with a schema within our framework. First, it can become directly associated with the schema itself, for example hearing a word like *'grasp'* may become associated with a schema that represents the action of grasping objects, the context in which it is applicable and the result of having grasped an object. Second, an observation can become associated with another observation, which may appear in multiple schemas. For example, the word *'red'* may become associated with any visual observations containing red objects.

3.3   Schema chaining

The linking of pre-conditions and post-conditions from different schemas (*'schema chaining'*) creates a traversable network representing different world states and the actions required to move between these states, as illustrated in figure 3.8. Without schema chaining the robot's interest in unreachable objects would decrease as it failed to reach them. Schema chaining allows for cases in which the feedback of an action isn't instantaneous to still be recognised as being useful. Thus making the entire series of actions required to point at an object, wait for another agent to move the object and then reach out to touch the object interesting to the robot.

In our implementation schema chaining is achieved by applying Dijkstra's algorithm [29] to this network of world states. We treat each state as a vertex in the network of potential world states and each action as an edge between two states. These paths are then weighted based on the probability of success, allowing the system to determine the shortest chain of actions required to achieve a goal in the manner most likely to succeed.

The potential states that make up the vertices are determined by matching the post-conditions of the last schema tested against the pre-condition of the next

Pre-conditions	Action	Post-conditions
Object out of reach	Point to object	Object in reachable area

Pre-conditions	Action	Post-conditions
Object in reachable area	Reach to object	Object in reachable area Touching object
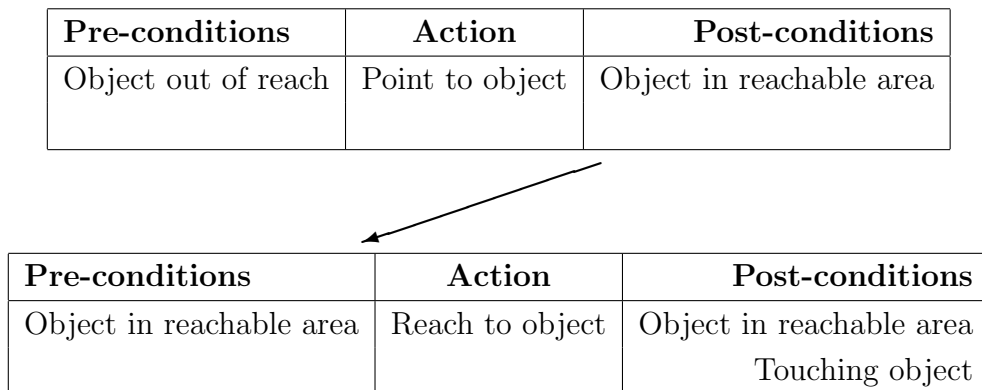
Figure 3.8: A high level example of schema chaining, allowing the robot to gain access to an object that would otherwise be outside of its reach through communication with another agent.

candidate. Any generalised schemas have their values populated based upon the prior schema in a potential chain before being tested for compatibility.

During the execution of a chain the robot evaluates the success of the predictions made by each schema in the chain after the action component of that schema has been executed. If the execution of one of the schemas results in an unexpected outcome which is not compatible with the next schema's pre-conditions then the chaining process begins anew, finding a chain of schemas from the new world state to the target.

Continual evaluation of the environment during the execution of a chain allows the system to adapt to both incorrect or unreliable learning and to other agents acting upon the environment. This is investigated further in chapter 5.8, in which the robot makes use of this functionality to cope with interference from a third party.

3.4   Schema excitation

To determine which schema should be executed next we make use of an intrinsic motivation system, focusing on the novelty of experiences [102]. 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 aimed at expanding its own knowledge of the world.

For the purposes of illustration we consider a robot which has already learnt a number of sensorimotor patterns relating to looking at its own hand and moving its hand to the location it is looking at. If we now introduce a previously unseen object the robot's memory of seeing its own hand in the location the object now occupies is partially activated based on the similarity of the two experiences. In the absence of more exciting stimuli this is enough to cause the robot to perform the action it remembers being associated with seeing its hand in that location, namely moving its hand into that location. The final result being that upon being presented with a new object the robot will reach out and touch the object and so gain further novel experience.

By combining the novelty of the current experience with the similarity to past experience a behaviour emerges that results in the robot selecting actions likely to be relevant to the novel components of new experiences.

3.4.1   Implementation

A schema's excitation level is found by first comparing each observation present in the current world state with all the post-conditions combined with observations associated with the schema as a whole and observations associated with components within that schema.

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 end effector (a green touch sensitive 'finger' in the case of the Adept robot, and a yellow hand in the case of the iCub) 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. 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.

When calculating the excitation from an observation associated with a component of a schema, the ratio of the number of times a remembered associated observation and the observation it was associated with are seen together and the number of times the associated observation is seen in other contexts is used to weight the excitation contribution from that observation. When considered in the context of language learning this results in words that are heard coincidentally in a large number of different contexts having a much lower weighting when compared to words which are more tightly related to a specific element of the scenario. The effects of this can be seen in chapter 5.10, an experiment in which we introduce words such 'grasp' and 'drop', which relate directly to actions the robot has taken, alongside words like 'this' and 'is', which can occur in a wide variety of different contexts.

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 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. A schema which can be executed immediately has a distance value of 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. In this case the schema is given an excitation value of 0.

The excitation value of the schema as a whole is then diminished by the number of times that schema has been activated, this allows schemas that are

not producing new information (and so new schemas) to become less interesting and make way for other schemas to be investigated in the current context. If the schemas which appear relevant to the current world state are not producing any new results the robot becomes 'bored' with them, eventually resulting in random execution of barely related schemas in the hope of eliciting some new response from the world.

A simple optimisation when determining whether or not a schema is reachable via a chain is to keep track of the current highest excitation value of previously tested schemas. If the schema being evaluated can't be executed directly and its estimated excitation is less than half that of the highest previously tested schema then it is not necessary to attempt to find a chain leading to this schema. This is because it would never be selected as the most exciting schema even if the path selected only had two steps, since the excitation is divided by the length of path required to reach it. This optimisation is not implemented here, as for experimental purposes it can be valuable to obtain the excitation values of all schemas, not just those executed. Additionally while the implementation used in the following experiments is single threaded, the process itself can be executed in parallel, allowing for the excitation of each schema to be evaluated in a separate thread, making more efficient use of multi-core processors.

The full procedure for calculating the excitation of a schema based upon the current world state can be seen in algorithms 1 and 2.

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

Algorithm 1 Calculate the excitation of a schema in the context of the provided world state.

---

    function EXCITATION($schema, worldstate$)

        $excitation = 0$

        if $schema$'s pre-conditions are a subset of $worldstate$ **then**

            $distance = 1$

        else

            if $schema$ is unreachable **then**

                return 0

            end if

            $distance =$ length of path to $schema$

        end if

        for each $observation1$ in $worldstate$ **do**

            for each $observation2$ in $schema$'s post-conditions **do**

                $excitation \mathrel{+}=$ similarity($observation1, observation2$) / number of times $observation1$ has occurred

                for each $associatedObservation$ associated with $observation2$ **do**

                    $excitation \mathrel{+}=$ similarity($observation1, associatedObservation$) * number of times $associatedObservation$ and $observation2$ have occurred together / number of times $observation1$ has occurred

                end for

            end for

            for each $associatedObservation$ in $schema$'s associations **do**

                $excitation \mathrel{+}=$ similarity($observation1, associatedObservation$) * number of times $associatedObservation$ seen alongside $schema$ / number of times $observation1$ has occurred

            end for

        end for

        $excitation = excitation$ / (number of times this schema has been executed * $distance$)

        return $excitation$

    end function

63

Algorithm 2 Calculate the excitation contributed by the similarity between two observations.

---

    function SIMILARITY($observation1, observation2$)
        if $observation1$ and $observation2$ are of the same type **then**
            $similarity = 0.25$
        else
            return 0
        end if
        for each $property1$ in $observation1$ **do**
            if $observation2$ has property with the same name and value as $property1$
then
                $similarity \mathrel{+}= 1$ / total number of properties in $observation1$
            end if
        end for
        return $similarity$
    end function

3.5   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 schema's pre-conditions then the new outcome is added to that existing schema and the probability of it occurring is tracked. An illustration of this process can be seen in figure 3.9.

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 ∪ Predicted post-conditions
Object in field 4
Finger in field 4

$\Downarrow$

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

Figure 3.9: An example of the process leading to a new schema being created.

3.6   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.

This generalisation mechanism produces schemas containing parameters (or *'slots'*) which can be populated based upon the current experiences of the robot when being executed. Beyond simply determining which aspects of the schema may be interchangeable with other values as many existing schema systems do, this mechanism attempts to find generalisable relationships between the pre-conditions, the action and the post-conditions of a schema. This allows the generalised schemas which are produced as a result of this process, to represent the agent's hypotheses about how an interaction may work at a more abstract level.

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 resulting post-conditions of the found schema must be a subset of the post-conditions of the schema upon which generalisation is taking place. The post-conditions of the found schema are then used to convert the action of the schema being generalised 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 3.10.

If the action has no parameters then this stage is unnecessary. Although this is not strictly necessary when the action's parameters are of a compatible type with the observations within a schema it can still be beneficial to make use of a target

action, as this allows the generalised schema to make use of any adjustments made by the simpler schemas which fulfil its target. For an example of this see chapter 5.2.1.

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	*Target:* 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.

Figure 3.10: An illustration of the process for forming a target action.

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 3.11.

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. We determine whether a generalised schema is reliable enough based upon two simple dynamic thresholds which take into account the reliability of prior generalisations. If earlier generalisations are proving to be unreliable then new ones must be tested against a greater number of similar schemas and achieve a higher rate of success. We do this by considering the relationship between the number of times that any generalised schemas have been executed (Υ) and the number of times these generalised schemas have successfully predicted the outcome of the action undertaken (Ω). The formulas for these thresholds are expressed in equations 3.1 and 3.2.

$$evidenceThreshold = \begin{cases} 1 + \Upsilon & \text{if } \Omega \text{ is } 0 \\ 1 + \frac{\Upsilon}{\Omega} & \text{otherwise} \end{cases} \tag{3.1}$$

$$satisfactionThreshold = \begin{cases} 0.5 & \text{if } \Upsilon \text{ is } 0 \\ 1 - \frac{\Omega}{2\Upsilon} & \text{otherwise} \end{cases} \tag{3.2}$$

The full details for generalisation of schemas can be viewed in algorithms 3, 4, 5, 6 and 7. The values *evidenceThreshold* and *satisfactionThreshold* are populated based upon the formulas described in 3.1 and 3.2 respectively.

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.

When resolving a target action we can make use of the schema chaining mechanism previously defined in section 3.3, since this takes as its input the current world state and a target world state and outputs either a single schema or a chain of schemas which will achieve this target state. This results in the schemas developing a hierarchy in which more basic schemas can be utilised as components in generalised schemas. This differs from the form of hierarchy seen in many other schema based works, which focus on the creation of a hierarchy of schemas by encapsulating schema chains within new individual schemas. While this is also theoretically possible within our own framework that aspect of schema hierarchy is not investigated here.

Pre-conditions	Action	Post-conditions
Object in field 4	*Target:* Finger in field 4	Object in field 4 Finger in field 4 Touching

↓

Pre-conditions	Action	Post-conditions
Object in field $x	*Target:* Finger in field $x	Object in field $x Finger in field $x Touching
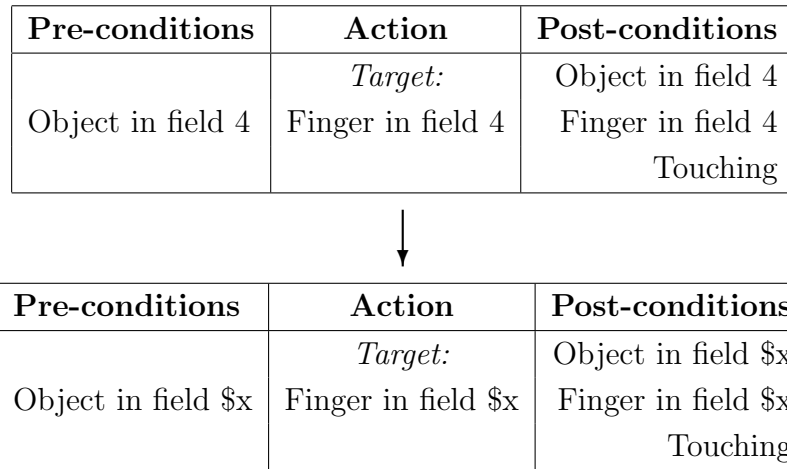
Figure 3.11: A schema with its concrete action replaced by a target action can then be converted into a generalised schema.

When executing a target action it is important that the parent schema of that action is removed from the pool of potential schemas that may be used to resolve the target. Otherwise a scenario can emerge in which a target action resolves itself using its own parent schema, resulting in an infinite loop as illustrated in figure 3.12

Pre-conditions	Action	Post-conditions
Object in field $x	*Target:* *Finger in field $x*	Object in field $x Finger in field $x Touching

↓

Pre-conditions	Action	Post-conditions
Object in field $x	*Target:* *Finger in field $x*	Object in field $x **Finger in field $x** Touching

↓

Pre-conditions	Action	Post-conditions
Object in field $x	*Target:* *Finger in field $x*	Object in field $x **Finger in field $x** Touching

↓

Ad infinitum
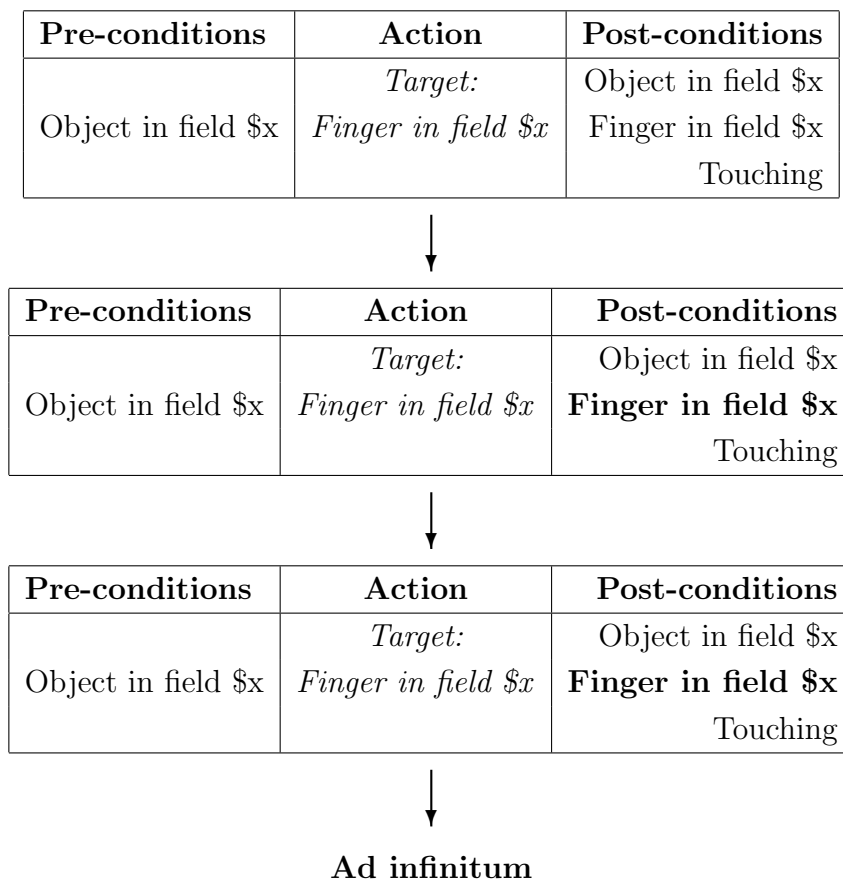
Figure 3.12: When attempting to resolve a target action it is possible that the parent schema of that action may be selected as the solution most likely to succeed, as that schema will always contain the desired target state. For this reason it is advisable to remove the parent schema from the pool of potential schemas prior to resolving a target action.

Algorithm 3 Schema generalisation through parameterisation.

    function GENERALISE($newSchema, existingSchemas$)
        // Find similar schemas and possible target action
        $minSize$ = number of pre-conditions in $newSchema$
        $targetUsed$ = true
        $action$ = a new target action
        for each *schema* in *existingSchemas* **do**
            if *schema* already generalised **then**
                continue
            end if
            if similar($newSchema$, $schema$) **then**
                Add *schema* to list of *similarSchemas*
            end if
            if $newSchema$'s action has no parameters **then**
                $action$ = a copy of $newSchema$'s action
                $targetUsed$ = false
            else if $newSchema$'s action is the same as *schema*'s action **then**
                // Find the result of this action in the simplest context
                if number of pre-conditions in *schema* less than $minSize$ **and** *schema* has post-conditions **and** satisfies(*schema*, $newSchema$) **then**
                    $minSize$ = number of pre-conditions in *schema*
                    $action$'s target = *schema*'s post-conditions
                end if
            end if
        end for
        if number of schemas in *similarSchemas* less than *evidenceThreshold* **then**
            // Not enough evidence to generalise this schema yet
            return
        end if
        if $targetUsed$ **and** $action$'s target is null **then**
            // Unable to find suitable target action
            return
        end if

Algorithm 3 Schema generalisation through parameterisation (continued).

   for each *schema* in *similarSchemas* **do**
      // Find different generalisations possible from each similar schema
      variables = findVariables(*schema*)
      if *variables* is empty **then**
         return
      end if
      trialSchema = copy of *newSchema*
      trialSchema's action = copy of *action*
      usedVariables = generaliseState(reference to *trialSchema*'s pre-conditions, *variables*)
      if *targetUsed* **then**
         usedVariables = *usedVariables* + generaliseState(reference to *trialSchema*'s target action, *variables*)
      end if
      generaliseState(reference to *trialSchema*'s post-conditions, *usedVariables*)
      if generalisation matching *trialSchema* already exists **then**
         continue
      end if
      satisfied = 0
      for each *testSchema* in *similarSchemas* **do**
         if satisfies(*testSchema*, *trialSchema*) **then**
            Increment *satisfied*
         end if
      end for
      if (*satisfied* / number of schemas in *similarSchemas*) less than *satisfactionThreshold* **then**
         continue
      end if
      Add *trialSchema* to the schema memory.
   end for
end function

Algorithm 4 Determine whether two schemas are similar for the purposes of generalisation.

---

    function SIMILAR(*schema*1, *schema*2)

        if number of pre-conditions and post-conditions in *schema*1 differ from those in *schema*2 **then**

            return false

        else if *schema*1's action type is different to *schema*2's action type **then**

            return false

        else

            for each *observation*1 in *schema*1's pre-conditions **do**

                if *observation*1 not contained in *schema*2's pre-conditions **then**

                    return false

                end if

            end for

            return true

        end if

    end function

Algorithm 5 Generalise a world state using a provided set of candidate variables.

---

    function GENERALISESTATE(reference to *state*, *variables*)

        usedVariables = empty hash map with variable names as keys

        for each *observation* in *state* **do**

            for each *property* in *observation* **do**

                for each *var* in *variables* **do**

                    if *var*'s value equals *property*'s value **then**

                        Set *property*'s value to *var*'s key

                        Add *var* to *usedVariables*

                  end if

                end for

            end for

        end for

        return *usedVariables*

    end function

Algorithm 6 Find potential variables for parameterisation of schemas when generalising.

---

function FINDVARIABLES(*schema*)

    variables = empty hash map with variable names as keys

    for each *observation*1 in the union of *schema*'s pre-conditions and *schema*'s target action **do**

        for each *observation*2 in *schema*'s post-conditions **do**

            for each *property*1 in *observation*1 **do**

                if *property*1 doesn't exist in another schema's pre-conditions within *similarSchemas* **then**

                    continue

                end if

                if *observation*2 doesn't have a property with the same name as *property*1 **then**

                    continue

                end if

                *property*2 = property from *observation*2 with the same name as *property*1

                if value of *property*1 equals value of *property*2 **then**

                    if value of *property*1 doesn't already exists in one of *variables*' values **then**

                        varname = Previously unused variable name

                        Add variable to *variables* using *varname* as the key and *property*1's value as the value

                    end if

                end if

            end for

        end for

    end for

    return *variables*

end function

Algorithm 7 Determine if schema1 is able to achieve the same result as schema2 given similar starting conditions

---

function SATISFIES(*schema*1, *schema*2)

    if *schema*1 is generalised **then**

        Instantiate *schema*1's variables based on the values in *schema*2's pre-conditions

    end if

    if *schema*2's pre-conditions are a subset of *schema*1's pre-conditions **then**

        if *schema*2's post-conditions are a subset of *schema*1's post-conditions **then**

            return true

        end if

    end if

    return false

end function

3.7   Generalisation of associated observations

When associating observations with other observations (as opposed to associating observations with entire schemas), a similar approach to that taken for schema generalisation can be used, allowing these associations to ignore properties within an observation which are unrelated to the current association.

Figure 3.13 shows an example in which two visual observations have both become associated with the robot hearing the word *'red'*. Given a number of similar associations in which some properties remain constant while others change, the generaliser assigns variables to the other changing properties which can then be instantiated based on current observations when wishing to determine if this generalised association applies to any elements of the current world state. The remaining concrete property values are then assumed to be the elements of this observation which actually relate to the associated observation. In this example the result is that the colour and object ID properties become associated with the word *'red'*. In this case the object ID is becoming associated because objects are identified based on their colour by the vision system, so in all examples encountered the robot will find that object ID 1 occurs alongside the colour red.

The implementation of this mechanism can be seen in algorithm 8.

Visual observation
Colour: Red
Object ID: 1
X: 78.32
Y: 12.43

Auditory observation
Word: RED

Visual observation
Colour: Red
Object ID: 1
X: 52.93
Y: 14.72

Auditory observation
Word: RED

$\Downarrow$

Visual observation
Colour: Red
Object ID: 1
X: $a
Y: $b

Auditory observation
Word: RED
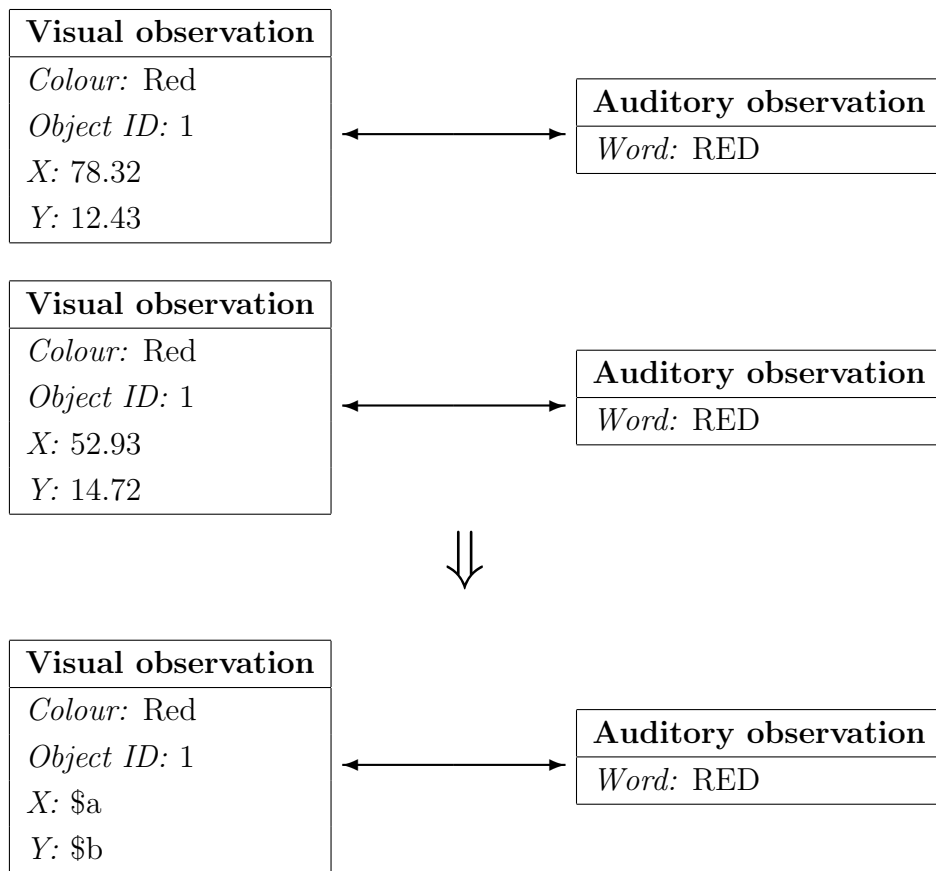
Figure 3.13: The creation of a generalised association based on two concrete associations.

Algorithm 8 Generalise the properties of an association between two observations.

function GENERALISEASSOCIATION(*newAssociation*, *existingAssociations*)

    for each *association* in *existingAssociations* **do**

        if *association*'s first observation is of the same type as *newAssociation*'s first observation **and** *association*'s second observation is equal to *newAssociation*'s second observation **then**

            for each *prop* in *association*'s first observation **do**

                if *newAssocation*'s first observation has a different value for property with same name as *prop* **then**

                    if *prop* not already in *differentProperties* **then**

                        Append *prop* to *differentProperties*

                    end if

                end if

            end for

            Append *association* to *similarAssociations*

        end if

    end for

    if *similarAssociations* has less than 2 associations **then**

        return

    end if

    generalisedObservation = copy of *newAssociation*'s first observation

    if *differentProperties* is empty **or** *differentProperties* contains the same number of properties as *generalisedObservation* **then**

        return

    end if

    for each *property* in *differentProperties* **do**

        Replace property corresponding to *prop* in *generalisedObservation* with unused variable

    end for

    generalisedAssociation = new association between *generalisedObservation* and *newAssociation*'s second observation

    if *generalisedAssociation* doesn't already exist **then**

        Add *generalisedAssociation* to schema memory

    end if

end function

3.8   Schema based observation

The schema system is capable of being used as a filter through which the actions of others can be interpreted and given meaning. We achieve this by considering the state of the world prior to another agent acting and then again after they have completed an action. We then look for a schema with similar pre-conditions and post-conditions, from this we can infer what action we would have taken to achieve the same result.

This knowledge could then potentially be used for training another component of the system to recognise visual properties of the other agent's action and classify them according to our own representation of that action. This would allow us to then recognise that action outside of contexts that we're already familiar with. Such training of visual recognition sub-systems is outside the scope of this investigation but is discussed in more detail in chapter 6.2.5 (Further work).

We can take this a step beyond simply learning the correspondence between others' actions and our own to allow us to predict their future actions and eventual goal. We achieve this by keeping a record of schema chains that we have previously found to be useful and then searching for the chains in which the action we have just observed occurs. From this we can form one or more hypotheses as to the other agent's next action, with each action further narrowing the list of potential schemas until we arrive at a single prediction as to their desired goal. This provides the capacity for the schema learning framework to perform plan recognition based upon its own experiences of the world.

An example of this predictive capability is seen in experiment 5.14.

Chapter 4

Robotic Platforms

Experimentation was carried out across two robotic platforms. An Adept six axis manipulator arm was used for experiments based around touching and pointing, while an iCub humanoid robot was used for similar touching and pointing experiments along with additional object interaction experiments. Experiments on both platforms involved aspects of learned communication and intrinsic motivation.

4.1   Adept manipulator arm

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 and so allowing the robot to point 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 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 4.1.

4.1.1 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 simulation 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 body physics, allowing for semi-realistic interactions between the arm and its environment. This simulation environment can be seen in figure 4.2. 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.

4.1.2 Pointing mechanism

The controller takes the system through two learning stages to create a mapping between the motor system and the vision system. This mapping allows the robot to move its end effector into a desired visual field, which can then be used for allowing it to interact with objects (both by physically touching them and by pointing at them for communication).

The first stage of this process is akin to Piaget's first stage infant, the robot goes through a period of *'motor babbling'*, where it exercises all possible joint configurations and creates schemas representing these actions. It receives no feedback from these actions, merely generating a base set of schemas that abstract higher level schemas away from explicit joint commands, allowing them to instead refer to existing schemas as their action components.

In the second stage the vision system is made available to the robot and it begins to associate visual context with the existing motor schemas. This is similar to hand fixation in an infant. This stage is visualised in figure 4.3. The robot

Figure 4.1: The Adept arm system and camera.
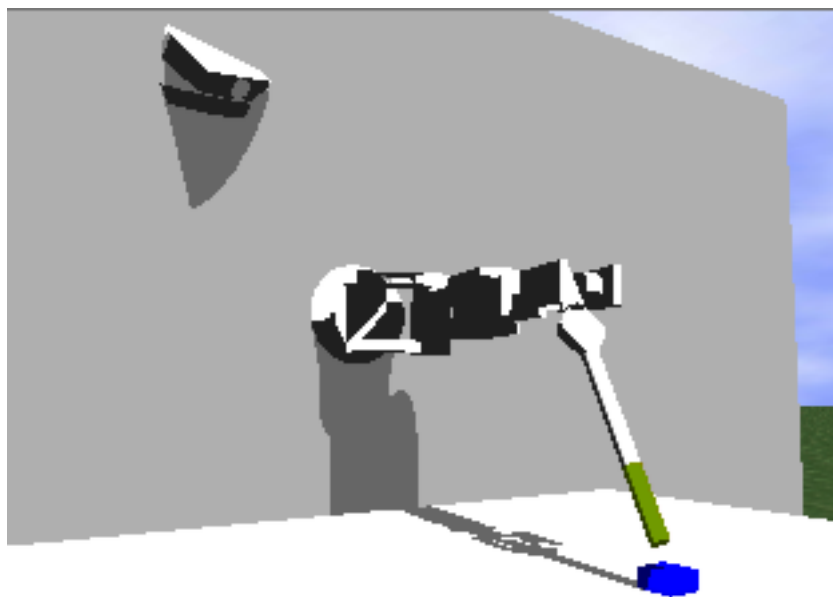
Figure 4.2: The simulation environment showing the arm pointing at an object placed slightly outside of the robot's work envelope.

executes the purely motor based schemas it has learnt in the previous stage and forms a new visual field whenever it sees its end effector outside of any existing fields, it then adds this as a new post-condition to the executed schema. The end effector is detected via the vision system, potentially it will add any changes in visible objects as post-conditions, however at this stage of the robot's learning no other objects are presented to it.

The system operates primarily on the X-Y plane using 2 degrees of freedom, illustrated in figure 4.4(a). To enable the robot to point at objects outside of its work envelope it is able to slightly lift its end effector when at the furthest extent of its normal range of motion, shown in figure 4.4(b). Both of these planes are accessible to the robot throughout all stages of learning, so it first learns to position its end effector in the 'pointing' plane prior to any objects being introduced for it to point at as part of its random motor babbling and vision mapping stages.

It is important to note that this is not giving the robot a full 3D representation of the space it occupies as the robot still effectively lacks accurate depth perception, the iCub system discussed later does contain the ability to judge

Figure 4.3: A visualisation of the visual fields, part-way through their discovery.

the distance of objects and the differences in representation that this causes are investigated in chapter 5.3.

A similar system has since been investigated by Hafner and Schillaci [49]. In their experiments they extended this mechanism of learning proto-imperative pointing through failed grasping to work with a 3D reach space.

4.1.3   Morphological implications

This approach raises certain morphological implications. For a pointing gesture that a human would recognise to emerge from this technique the robot in question must itself have a roughly humanoid anatomy. Specifically it requires the robot's vision system to be positioned above the arm system looking out in the direction of action. Additionally for the pointing to appear accurate the vertical distance between the vision system and the arm should not be too great.

All current testing has been performed with humans with prior knowledge that what they are about to view is intended as a pointing gesture, it might be interesting to investigate the effects this gesture has on people who do not already know what to look for. The anthropomorphic characteristics of the robot in question might play as large a part in this as the quality of the gesture itself. However this is outside the scope of the current investigation.
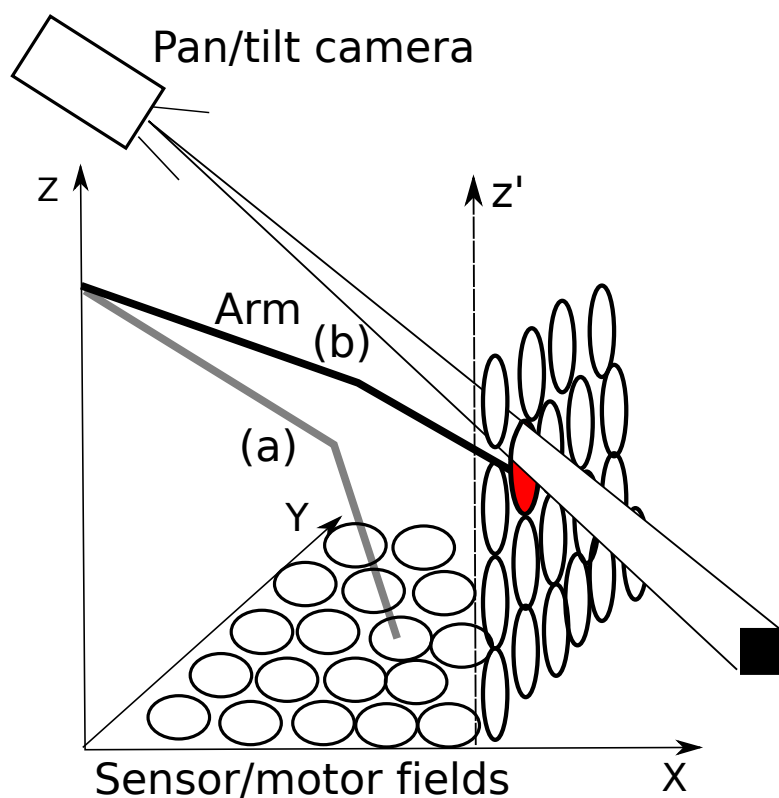
Figure 4.4: *(a)* Direct contact with objects is only possible in the X-Y plane. *(b)* The robot's motion is extended into the Z' plane allowing it to point at distant objects. This is a simplified example, rather than having two distinct vertical and horizontal planes the system operates on a manifold that curves up at the extremities tracing the outside of the robot's work envelope.

4.1.4 System architecture

The Adept system is based around the Player robotics framework. The Player server communicates with either the physical robot or the simulated one, vision from either system is passed through a blob detector before being sent to the developmental controller along with information on touch responses. Audio input occurs outside of Player, being handled directly by the developmental controller. CMU Sphinx is used for speech recognition with a limited dictionary of predefined words.

The developmental controller then makes use of the PSchema library[1] which implements the schema learning functionality detailed in the previous chapter. In this manner PSchema itself is independent of a specific robotics framework or platform, nor is it necessarily tied to robotics at all.

Unlike the iCub architecture, detailed in the following section, the developmental controller operates directly in terms of joint configurations, with no intermediary sensorimotor maps being created. Instead all mapping takes place within the schema memory, during the initial learning phase the robot goes through a period of motor babbling in which the arm is placed in a number of different locations, schemas are then created mapping the action in joint space to the visual result, this initial learning is discussed further in section 5.1. An example of a schema produced as a result of this process can be seen in figure 4.5.

Pre-conditions	Action	Post-conditions
	Joint configuration -1.57, 2.61	Finger in field 7

Figure 4.5: A schema produced during the early stages of learning on the Adept robot, mapping a specific joint configuration to its visual result.

Figure 4.6 illustrates the components of this architecture and their connections with each other.

---

[1] A general purpose schema learning library, developed as part of this investigation and made publicly available at `http://pschema.org` under the GNU General Public License (GPL).
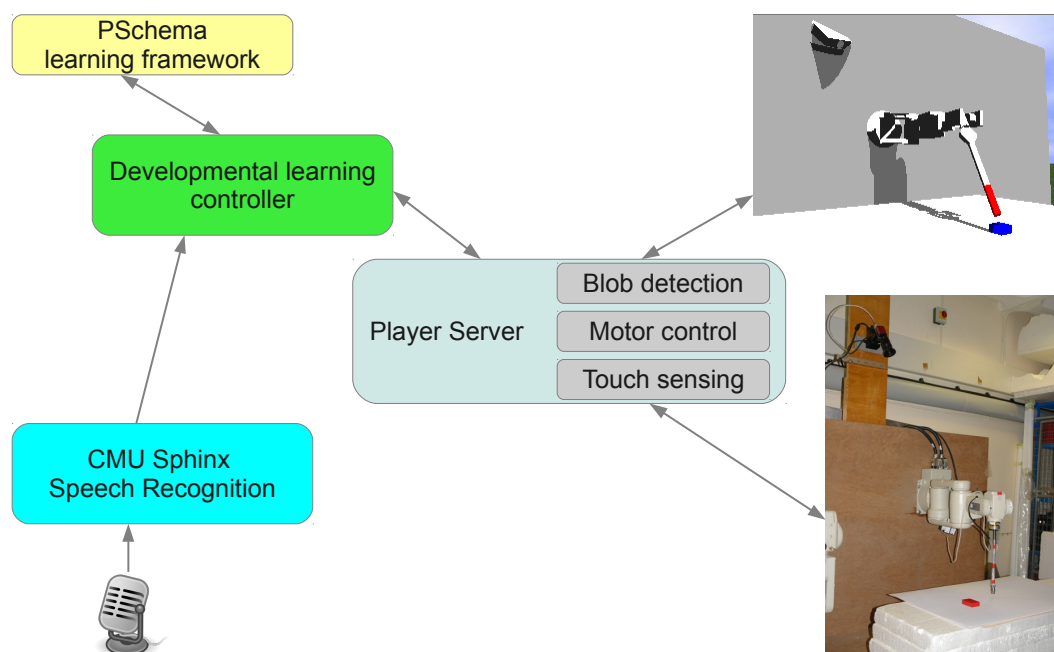
Figure 4.6: A high level overview of the Adept system architecture.

4.2   iCub humanoid robot

Figure 4.7: The iCub humanoid robot.

The iCub, shown in figure 4.7, has been developed as a common robotics research platform as part of the RobotCub project [72, 98]. In total it provides 53 degrees of freedom, however in our experiments we do not make use of this full range. The components made use of here are the head, one arm and one hand. Together these provide 22 degrees of freedom, 6 of which are distributed between the neck and eyes in the head, 7 control the arm and a further 9 control the hand.

The hand has a triangular array of 108 touch sensors, while the upper arm provides 6-axial force/torque feedback. Due to feedback issues resulting in poor reliability and safety issues the touch sensors are only used for evaluating grasps and are not used to determine if the robot has come into contact with an object

when reaching towards it. Instead the robot is required to always reach slightly above an object to account for the iCub's wrist tendon being likely to become damaged if the hand is flexed upwards as would happen when reaching down on to an object. The sensation of touching an object in the following experiments is simulated within the sensorimotor controller whenever the hand is perceived to be in the same visual location as an object. This also necessitates that a small amount of assistance is provided to the robot when grasping objects, as they must be lifted slightly for the hand to be able to reach them.

Two cameras mounted in the head provide a stereo vision system, each with a resolution of 640x480 pixels at 15 frames per second. This stereo vision is then used to provide rough depth estimates.

The iCub is capable of displaying expressions through a number of LEDs underneath the face plate. These can be illuminated in various configurations to achieve displays such as surprise, happiness, confusion, sadness, or anger. This functionality is used to provide a rough indication of the robot's internal state by showing a look of concentration whilst performing an action followed by a happy or sad expression if the action was successful or not.

4.2.1   Simulated robot

The iCub simulator, shown in figure 4.8, provides a 3D environment with simulated physics. All the functionality of the real robot is simulated, allowing for a full range of movement, stereo vision, touch sensing and facial expressions.

While the simulator allows the robot to interact with objects there are some limitations. The simulation of the fingers is not fine enough to be able to accurately grasp objects like we do with the real robot, instead a virtual grasping function is employed in the simulator which attaches nearby objects to the robot's hand.

As with the Adept system the same software is capable of driving either the physical robot or the simulated robot with only minor configuration changes.

4.2.2   System architecture

While the Adept system made use of the Player middleware system the iCub uses YARP. As with Player this provides the capacity for the same control software to
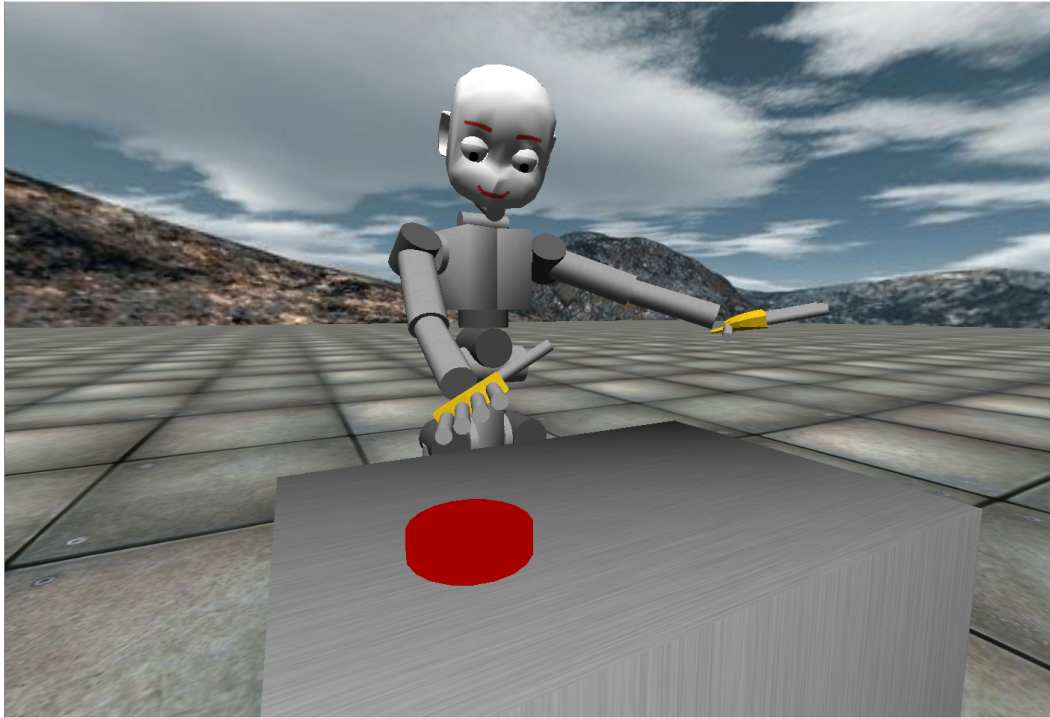
Figure 4.8: A 3D simulation of the iCub.

work with either the physical robot or a simulated environment without additional modification.

Rather than controlling the motor joints directly the developmental controller communicates with the sensorimotor controller (SMC) developed as part of the IM-CLeVeR project. The many additional degrees of freedom provided by the iCub make the use of direct joint configurations within the schema learning system impractical. Instead the schema system handles higher level control, while the SMC deals with the low level details involved in arm, head, eye and torso movements. This is illustrated in figure 4.9.
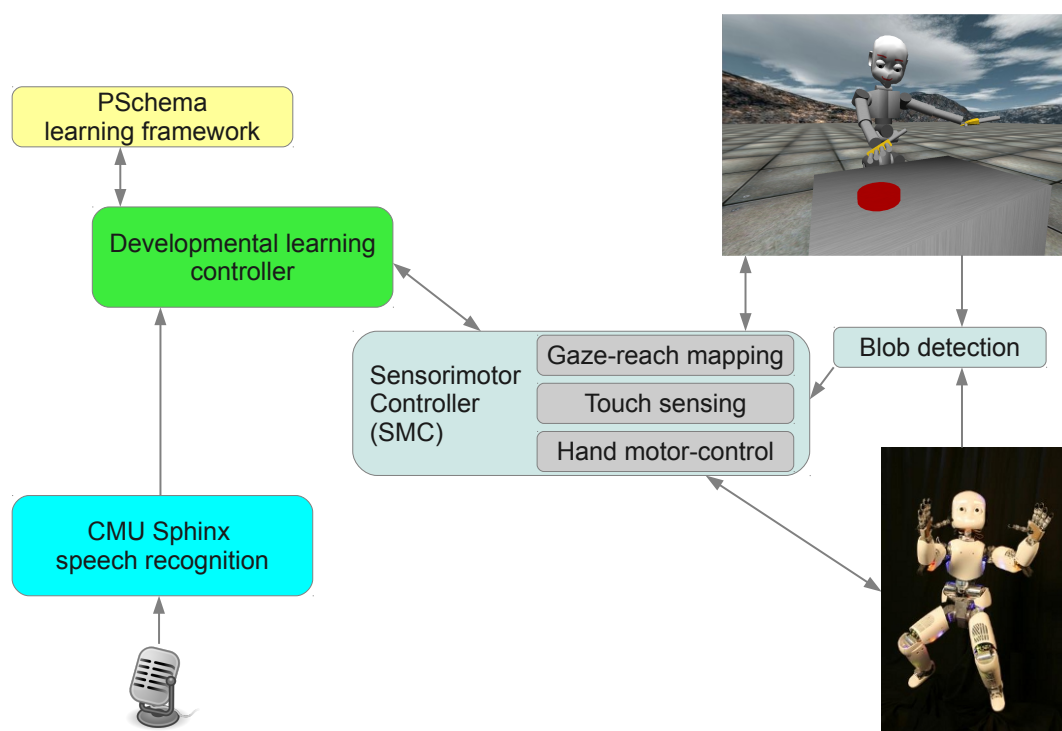
Figure 4.9: A high level overview of the iCub system architecture.

The SMC performs a similar mapping stage to the schema system on the Adept platform. However instead of just mapping the arm movements to visual results it creates an overall gaze space that co-ordinates both eye, head and torso movements and then creates a mapping between this and the arm system [52, 62]. A visual field will remain consistent within the gaze space even when the head or eye move to focus on a new object. The developmental controller then communicates with the SMC in terms of the combined gaze/reach space fields. Object and hand detection is performed by a colour based blob detection component within the SMC.

Figures 4.10 and 4.11 show the learnt mapping between the retinal space and the eye motor space. Circles of the same colour indicate a link between a given retinal position and the motor adjustment required to foveate upon it.
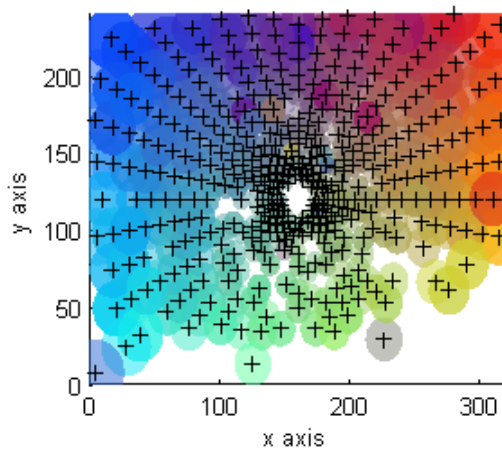
Figure 4.10: Visual fields allowing for the discrete representation of visual stimuli locations on the retina.
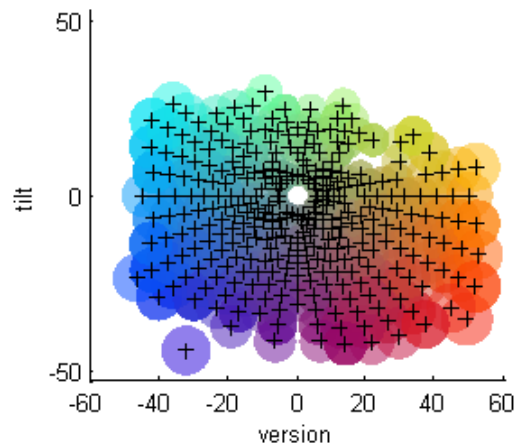
Figure 4.11: Eye motor fields representing the required motor adjustment to foveate upon a given field.

Initially only eye movement is possible within the motor system, however once these maps have reached a suitable level of coverage this constraint is lifted, allowing for the addition of head movements. Figures 4.12 and 4.13 illustrate the additional mappings which are learnt to account for the contribution of head movements towards foveation. This allows the robot to make large crude movements of the head which bring an object in to range so that the finer movements of the eyes can foveate upon it. Further learning of the eye mapping continues

during this stage if a visual stimulus is detected in a region not yet covered by the maps.
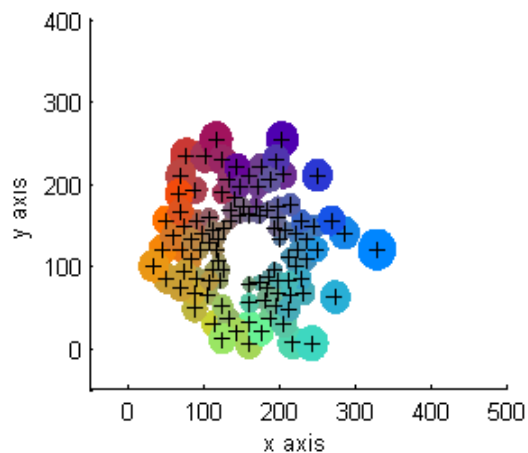
Figure 4.12: Visual fields representing regions in which visual stimuli have been detected, mapping to neck motor movements.
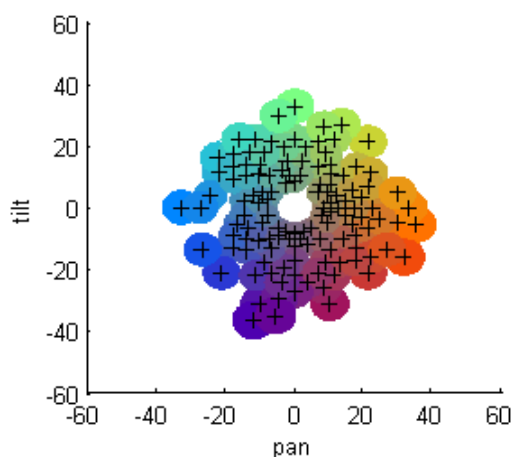
Figure 4.13: Head motor fields representing the neck movements required to bring a visual field in to range of the eye motors.

Finally a further constraint restricting the robot's ability to make use of its arm is lifted, allowing the robot to make movements with its arm and hand. As these movements are made a further set of maps are learnt between the visually detected hand location and the arm movements which placed it there, shown in figures 4.14 and 4.15. If a mapping does not exist between the visual location in which the robot's hand is detected and the head/eye motor system, linear piecewise interpolation based around the nearest known neighbours is used to approximate the motor movements necessary to foveate upon the hand. If this approximation does not correctly predict the movements needed then the actual result of this movement is recorded in a new mapping and a second attempt at interpolation is made using this additional information.

Through this combined set of maps it then becomes possible for the robot to fixate upon a visual stimulus and reach towards it.
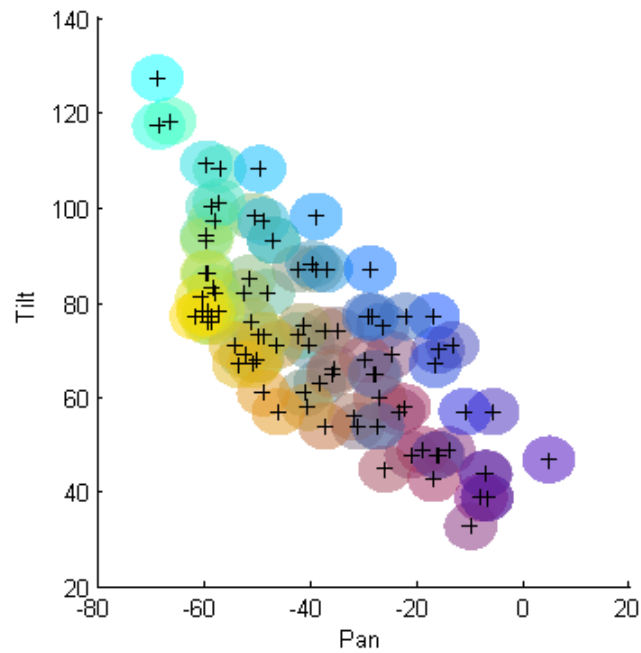
Figure 4.14: Visual fields in which the robot's own hand has been detected.
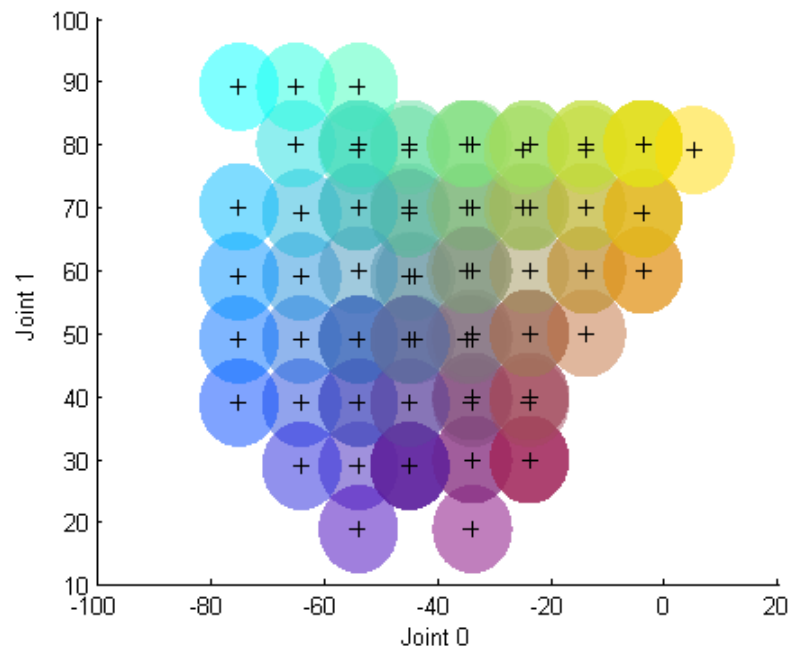
Figure 4.15: Arm motor fields representing the joint movements required to place the robot's hand in a corresponding visual field.

4.2.3   Pointing mechanism

As with the Adept, the iCub is capable of incidentally generating gestures which give the appearance of pointing when attempting to reach towards distant objects. The visual representation provided by the SMC is given in terms of the gaze direction of objects alongside an additional depth component calculated via the stereo vision system. By reaching as close as possible to an object along a given gaze direction a proto-imperative pointing gesture is produced.

The stereo vision system is used to produce a rough depth estimate and from this determine whether an object is within reachable range of the robot. This is exposed to the schema memory in terms of a binary reachability property, allowing the system to distinguish between objects that are within the robot's peri-personal space and objects that are within the robot's extra-personal space.

4.3   Time perception and the perception of other agents

While time segmentation has not been a focus of this work, it has been necessary to employ a simple mechanism for determining when one unit of time ends and another begins. Due to the action oriented nature of our framework we decided upon an egocentric action based interpretation of time, whereby each time step begins immediately prior to an action being performed and ends once all self-motion has stopped. This has advantages in terms of simplifying the problem of determining causal relationships between egocentric actions and their results, while this is beneficial for the majority of actions undertaken as part of the following experiments it introduces potential issues when other agents may be interacting with the environment independently of the robot's own actions.

The majority of actions taken by other agents within our experiments are as a result of social interaction with the robot and so can be accounted for by the robot's casual understanding of the world. In more complex scenarios in which other agents can also act independently of the robot a more sophisticated mechanism would be needed for segmenting their actions and then attributing the results of their actions to that agent instead of misinterpreting them as a result of the robot's actions.

While it is necessary to minimise the additional complexity of other agents acting upon the world during the robot's learning, once the schema memory has developed all the desired concepts it can operate in more complex and unpredictable environments. By making use of the continual evaluation of schema chains the robot can even adapt to other agents interfering with aspects of the environment relevant to the goal that it is currently working towards. This functionality is explored further in chapter 5.8.

Chapter 5

Experiments & Interpretation of Results

These experiments were carried out across the two physical robotic platforms and their two simulation environments, described in chapter 4. Different combinations of these platforms are used for different experiments depending on the capabilities required, however in a number of experiments we have attempted to duplicate them across both the Adept and iCub physical platforms to demonstrate that the learning mechanism can be usefully employed on very different hardware, with different overall system architectures.

The two platforms have a slightly different visual representation, on the Adept system visual information is given in terms of randomly numbered visual field IDs, whereas on the iCub this is represented as a pair of co-ordinates indicating the gaze direction which is calculated based on a combination of the head and eye configurations. The motor systems on each platform differ more significantly, on the Adept platform motor actions are given in terms of target joint positions, on the iCub platform reaching is done in terms of gaze co-ordinates which is made possible by the combined gaze-reach map learnt by the lower levels of that architecture. A more comprehensive description of the differences between the two physical platforms and their software architectures can be found in chapter 4.

On both platforms the visual location of the robot's own hand is automatically disregarded as a potential pre-condition. While this may be a reasonable assumption to make with regards to the hand a more comprehensive approach

which would encompass future similar scenarios by making use of an extension to the generalisation algorithm is discussed in chapter 6.2.1.

5.1 Initial learning

Prior to each of the following experiments the two platforms go through a period in which they exercise their most basic movements, enabling them to populate their schema memories with these actions and their results when performed without the presence of other objects or agents.

The same procedure is used in the corresponding simulation environments as on the physical robots prior to any experiments in which simulators were used.

5.1.1 Adept

The Adept arm moves through its range of potential joint configurations, moving each joint in 10 degree increments until it has fully explored its reach-space. After each movement it learns the visual effect of seeing the tip of its finger in each location. Due to the simplicity of this hardware platform no other action types are possible.

5.1.2 iCub

The iCub exercises all of the reaches supplied to it by the lower level sensorimotor controller (SMC), learning the resulting visual experience of seeing its hand in various different locations. In addition to these reaching schemas it also practices a pressing action, which has no effect, a grasping action which results in the robot receiving the sensation of touching its own hand due to no objects being present and a releasing action.

5.2 Experiment 1: Learning to touch

This experiment was carried out on both the Adept and iCub platforms. Each robot starts from having only performed the initial learning stage, having never previously encountered objects other than itself.

We begin by introducing a new object for the robot to interact with, the excitation of this new stimulus should cause the robot to be reminded of schemas related to the position the object occupies, resulting in it reaching toward the object and learning about touching objects. Once this has occurred we place new objects in different positions, giving the system the opportunity to form a generalised schema representing touching any objects in any positions.

5.2.1 Results

When the robot first sees an object it checks for schemas excited by that stimulus and finds that the most excited schema is one in which it remembers seeing its own hand in the location the object now occupies. This remembered schema is shown in figure 5.1.

Adept:

Pre-conditions	Action	Post-conditions
	Joint configuration -1.57, 2.61	Finger in field 7

iCub:

Pre-conditions	Action	Post-conditions
	Reach to 35, -66	Hand at 35, -66

Figure 5.1: The robot is reminded of a prior action involving the location the object now occupies.

Upon executing this the robot finds that when an object is present in the location it reaches its hand towards, it receives an unexpected touch sensation. A new schema, figure 5.2, is then formed to represent this knowledge.

With enough examples of this behaviour this can then be generalised into a form which represents reaching out and touching objects in any position, resulting in the generalised schema shown in figure 5.3. Despite the iCub's reach action being in terms of gaze co-ordinates, a target action is still generated as this allows the system to account for potential inaccuracies in the hand-eye mapping. For example, over time the calibration of the iCub can shift resulting in the mappings between the gaze space and the reach space becoming slightly out of alignment, given enough examples the schema system can adapt each of its low level reaching schemas to account for this. This adjustment occurs automatically as a result of the probability tracking performed on the post-conditions of actions,

Adept:

Pre-conditions	Action	Post-conditions
Object 1 in field 7	Joint configuration -1.57, 2.61	Object 1 in field 7 Finger in field 7 Touching

iCub:

Pre-conditions	Action	Post-conditions
Object 1 at 35, -66	Reach to 35, -66	Object 1 at 35, -66 Hand at 35, -66 Touching object 1

Figure 5.2: A new schema is created to represent the otherwise unpredictable knowledge the robot gained from this interaction.

as a new visual result of a reach action becomes more prevalent than the original the schema begins to predict that outcome in place of the original. This allows the system to gradually adapt to shifts of calibration on-line without requiring relearning of the original mappings, and by keeping the generalised schemas in terms of target actions these adjustments can be made use of by the higher level generalised schemas automatically.

Adept:

Pre-conditions	Action	Post-conditions
Object $a in field $x	*Target:* Finger in field $x	Object $a in field $x Finger in field $x Touching

iCub:

Pre-conditions	Action	Post-conditions
Object $a at $x,$y	*Target:* Hand at $x,$y	Object $a at $x,$y Hand at $x,$y Touching object $a

Figure 5.3: The creation of a generalised schema representing the act of touching objects in any location.

5.3 Experiment 2: Learning to point

This experiment was carried out on both the Adept and iCub platforms, due to differences in their capabilities we arrived at two slightly different representations,

as shown in the results section. For this experiment the iCub was provided with an extension to its visual perception allowing it to crudely estimate whether an object was reachable or not based upon its stereo vision system.

The robots began this experiment having already learnt how to touch objects from experiment 1, with the iCub this capacity was relearned to allow for the incorporation of the additional depth information provided in this experiment.
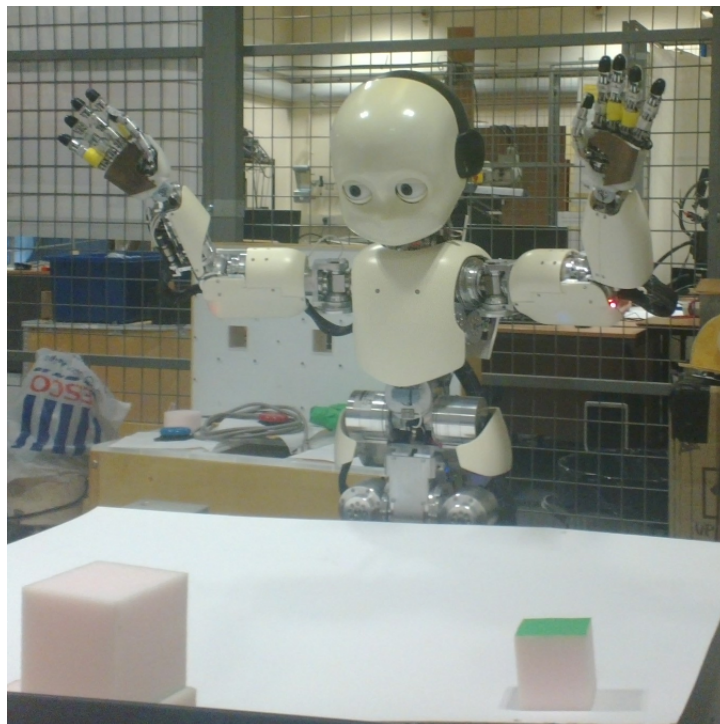
Figure 5.4: The iCub in its starting position with a green object placed in the robot's unreachable extra-personal space.

We first place an object outside of the robot's reachable work area (in its extra-personal space). When the robot attempts to reach towards the object it incidentally performs an action that looks to an outside observer like a pointing motion. When this occurs we move the object closer to the robot, as a parent might fetch a distant object that their child is reaching towards. Figure 5.4 shows the iCub with one example of an object placed outside of its reachable range.

5.3.1 Adept results

When presented with the object the Adept robot has no way of determining that it is out of reach and so the general touching schema (figure 5.5) is excited. This indicates that the robot believes that upon reaching out towards the object it should be able to actually touch the object.

Pre-conditions	Action	Post-conditions
Object $a in field $x	*Target:* Finger in field $x	Object $a in field $x Finger in field $x Touching

Figure 5.5: The Adept's generalised touching schema is excited by the distant object's presence.

However, upon completion of the action the robot finds that the object has been moved to a new reachable location and creates a schema representing this new knowledge, shown in figure 5.6. Although the finger appears in the same visual field that the object occupies, this is not the same physical location. As discussed in chapter 4.1.2, the projection of the distant object on to the robot's visual fields results in the appearance that the finger is in the same location, despite there being a large difference in the depth between the two physical locations.

Pre-conditions	Action	Post-conditions
Object 3 in field 87	Joint configuration 0.87, 2.26	Object 3 in field 54 Finger in field 87

Figure 5.6: A new specific schema is created representing the assistance the robot received from a third party moving an object when that object was pointed towards.

Due to the random, non-contiguous nature of the visual fields the Adept system must learn each pointing location individually, rather than creating a generalised representation of the pointing action. These are learnt as specific counter examples to the generalised touching schema.

5.3.2 iCub results

First the iCub is given the opportunity to learn a new touching schema, shown in figure 5.7, which makes use of the depth information provided by the iCub's stereo

vision. From this it learns that the general touching schema is only successful if an object is within its reachable work area.

Pre-conditions	Action	Post-conditions
Reachable obj. $a at $x,$y	*Target:* Hand at $x,$y	Reachable obj. $a at $x,$y Hand at $x,$y Touching obj. $a

Figure 5.7: The iCub is able to learn a more expressive touching schema which incorporates knowledge about the distance of an object.

When the iCub is presented with an object which is out of range the generalised touching schema doesn't get excited, unlike with the Adept. This is because the pre-condition of the object being reachable is not satisfied and no combination of actions currently exist to make it reachable. Instead a more basic reaching schema, figure 5.8, is activated which simply attempts to put the hand in the location the object currently occupies, with no prior expectation of the result beyond both the hand and the object being in the same place.

Pre-conditions	Action	Post-conditions
	Reach to 48.52, 16.03	Hand at 48.52, 16.03

Figure 5.8: The iCub is reminded of a simple reaching action involving the location the object now occupies.

From the perspective of an outside observer using a primitive reaching schema instead of a general touching schema provides the same visual appearance of the robot seemingly producing a crude pointing gesture. However the internal expectations of the robot are quite different, the Adept with its limited sensory information believes that it can touch the object, while the iCub is able to determine that the approach it has used previously for touching objects is not possible in this scenario. Instead the iCub selects the next most relevant schema, one which has no knowledge about objects but has a relationship with the position the object occupies, similar to the actions taken when first learning to touch nearby objects.

A specific schema representing pointing to this location is then created, figure 5.9, showing that if object 3 is out of reach and is in that specific location then

reaching towards it will result in the object being moved to another specific location. As with the Adept, the iCub's hand appears to be in the location that the object previously occupied, but is in fact at a different depth. This is due to both the object's initial position and the hand's final position lying in the same gaze direction.

Pre-conditions	Action	Post-conditions
Unreachable obj. 3 at 48.52, 16.03	Reach to 48.52, 16.03	Reachable obj. 3 at 14, 76.12 Hand at 48.52, 16.03

Figure 5.9: A schema representing pointing to a specific location.

Unlike with the Adept, if we then show a few additional examples with different objects in different locations a generalised schema can be formed which represents pointing to any distant location, shown in figure 5.10.

Pre-conditions	Action	Post-conditions
Unreachable obj. $a at $x, $y	*Target:* Hand at $x, $y	Reachable obj. $a at 14, 76.12 Hand at $x, $y

Figure 5.10: A generalised schema representing pointing to any location can be created by employing the depth perception available on the iCub.

While the resulting schema has most of its parameters generalised the final location that the object arrives at remains specific as this can not be predicted from any of the other values. In addition to this the final location can change in an unpredictable way, as this is selected by the human assistant when moving the object closer to the robot. The schema memory keeps track of all the different locations it has seen the object appear at and selects the most frequently experienced one as its prediction. The utility of this ability is investigated further in experiment 6.

5.4 Experiment 3: 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.

5.4.1 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 into a position that the robot cannot reach, however in attempting to touch the object it will form a pointing motion [101, 111] but will not receive a direct touch sensation, providing a counter example in which the generalised solution does not hold.

The placement of objects, with the exception of objects the robot has pointed towards, is done while the robot is not performing actions. For the purposes of schema creation the robot only evaluates the world immediately prior to and immediately after acting, so by only moving objects to new locations when the robot isn't acting we avoid the robot learning that objects always move when touched, which is a side effect of the experimental conditions rather than an actual property of the world.

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

5.4.2 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.

5.4.3 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.

5.4.4 Results

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

Table 5.1: The number of schemas produced in each experimental scenario.

Tables 5.1 and 5.2 summarise the number of schemas required to represent the scenario and the number of times an object must be moved to a new location in order to learn this representation.

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.

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

Table 5.2: The number of example object placements needed before a complete representation of each scenario can be achieved.

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 similar number of additional schemas would need to be added for every new object encountered by the system due to the lack of generalisation in this scenario. 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 is 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 precondition. 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. This flexibility is demonstrated to some extent by the fact that the same learning mechanism is capable of working with two different embodiments which have differing sensorimotor capabilities. Additionally, another solution to this problem is

possible through an extension to the generalisation algorithm, this enhancement is discussed in chapter 6.2.1.

5.5 Experiment 4: Learning to grasp

Due to the requirement for grasping objects within this experiment only the iCub was used. The robot begins this experiment with the knowledge gained during the touching experiment (5.2), meaning it already knows how to reach out and touch objects in any reachable location.

We begin by placing an object in front of the robot in a location at which it has previously never experienced an object, leading to this having high novelty and attracting the robot's attention. We then allow the robot to 'play' with the object, executing the most excited schema at each stage until it has learnt how to grasp the object.

5.5.1 Results

Pre-conditions	Action	Post-conditions
Object $a at $x,$y	*Target:* Hand at $x,$y	Object $a at $x,$y Hand at $x,$y Touching object $a

Figure 5.11: The generalised touching schema is excited by the presence of an object.

The touching schema, shown in figure 5.11, is executed a number of times due to the novelty of the experiences involved and their high relevance to the touching schema. However after a short while the excitation drops below that of the next most excited schema, which in this case is the grasping schema. The grasping schema, shown in figure 5.12, is excited by the memory of the robot touching its own hand when performing a grasp with no objects present, which it is reminded of by the touch sensation it receives from the object it has reached towards.

Executing this whilst touching an object results in the robot successfully grasping the object and receiving the sensation of holding an object. A new schema is then created to represent this new information, shown in figure 5.13.

Pre-conditions	Action	Post-conditions
	Grasp	Touching hand

Figure 5.12: A grasping schema learnt in the absence of objects is excited.

Pre-conditions	Action	Post-conditions
Object 1 at 35, -66 Touching object 1	Grasp	Object 1 at 35, -66 Hand at 35, -66 Holding object 1

Figure 5.13: A new schema is created to represent the unexpected effects of grasping when an object is present.

As with the new touching schema this grasping representation can also be generalised into the form shown in figure 5.14, which can represent the act of grasping an object in any location. In this case a target action is unnecessary as the grasp action has no parameters.

Pre-conditions	Action	Post-conditions
Object $a at $x, $y Touching object $a	Grasp	Object $a at $x, $y Hand at $x, $y Holding object $a

Figure 5.14: A generalised schema representing the act of grasping objects in any location once they have been touched.

Figure 5.15 shows the final state of the iCub at the end of the experiment, having successfully grasped the object.
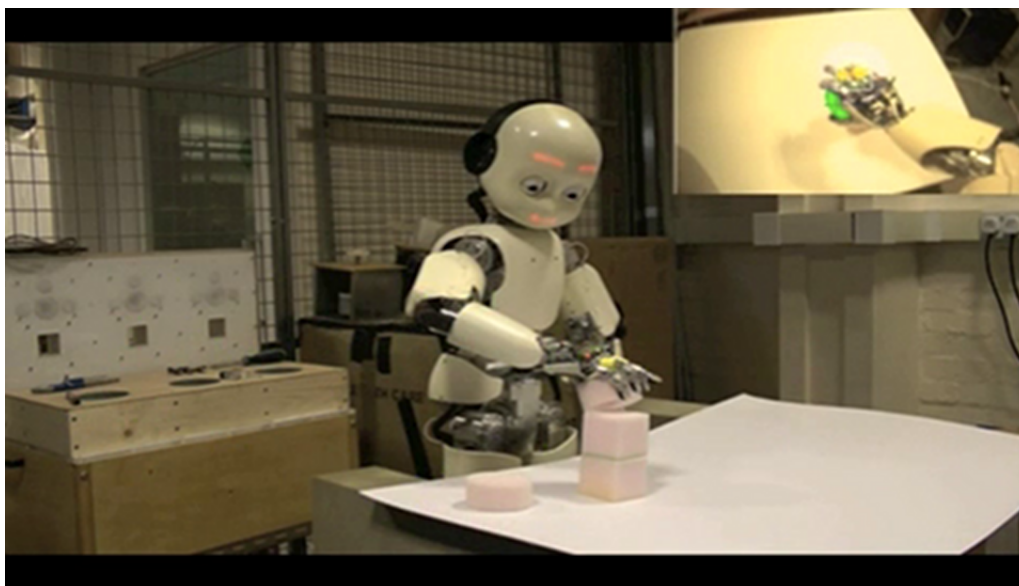
Figure 5.15: The iCub, having successfully grasped an object placed in front of it.

5.6 Experiment 5: Achieving target states

As discussed in chapter 3.3, the schema memory has the capacity to generate plans of actions by connecting the results of one schema with the requirements of another. In the following two scenarios we demonstrate this capacity and show how previously learned generalised schemas can be employed to generate plans when particular environmental configurations have not yet been directly experienced.

5.6.1 Scenario 1: Pointing

This experiment was carried out across both the iCub and the Adept robots.

Each robot begins with the knowledge gained from the earlier pointing experiment (section 5.3), with both robots having slightly different representations of what it means to point at an object resulting from their differing perceptions of the world.

We begin by placing an object outside of the robot's work area, however rather than allowing the system to execute the most excited actions as has been done in previous experiments, we explicitly request a target state from the system. For

this scenario our target is that the robot should be touching the object, as shown in figure 5.16.

Target world state
Touching object 3

Figure 5.16: Both robots are given the target of touching an object.

5.6.2   Adept results

Because the Adept has no general concept of pointing, it has learnt to point from a number of specific examples. One of these previously learned positions is chosen as the starting point for the object and the plan illustrated in figure 5.17 is generated. If the object were to be placed in a location that the robot had not yet had specific experience of it would fall back on its generalised touching schema, having no way of determining that the object is out of reach from its sensors, and so be unable to generate a valid plan.

When this plan is then executed the robot points towards the distant object and then with the assistance of a third party moving the object, the robot is able to reach out and touch the object in its new location.

5.6.3   iCub results

In contrast to the Adept the iCub has formed a generalised understanding of pointing based around its ability to determine whether an object is within its reachable space by using its stereo vision system to arrive at a rough depth estimate.

Figure 5.18 shows the plan produced by the iCub. Both the pointing and touching aspects of the chain are generalised, meaning that this plan could be constructed even were the object to be placed in some unknown out of reach location in which an object had never previously been encountered.

As discussed previously in section 5.3.2, the robot assumes that the object will be placed in a specific location after being pointed towards. We consider the implications of this further in section 5.7, where we demonstrate that despite the potential unreliability of the robot's assumption this type of chain can still be made practical use of.

Pre-conditions	Action	Post-conditions
Obj. 3 in field 87	Joint configuration 0.87, 2.26	Obj. 3 in field 54 Finger in field 87

Pre-conditions	Action	Post-conditions
Obj. $a in field $x	Hand in field $x	Obj. $a in field $x Hand in field $x, Touching

Figure 5.17: A specific pointing schema can be chained with the generalised touching schema to form a plan of action which allows the robot to ask for assistance in bringing an object close enough to be touched.

Pre-conditions	Action	Post-conditions
Unreachable obj. $a at $x, $y	*Target:* Hand at $x, $y	Reachable obj. $a at 14, 76.12 Hand at $x, $y

Pre-conditions	Action	Post-conditions
Reachable obj. $a at $x, $y	*Target:* Hand at $x, $y	Reachable obj. $a at $x, $y Hand at $x, $y Touching obj. $a

Figure 5.18: The generalised pointing and touching schemas can be chained together, allowing the iCub to request an object from any out of reach location and then touch it.

As with the Adept, when executed this plan correctly results in the robot pointing towards the distant object and then with the assistance of a third party moving the object, the robot is able to reach out and touch the object in its new location.

5.6.4 Scenario 2: Grasping

Due to the requirement for grasping in this experiment only the iCub has been used.

The robot begins in a state of having learnt how to both touch (5.2) and grasp (5.5) objects. We place an object at a previously unseen location on the table, In this scenario our target is that we want the robot to be holding the object we just placed in front of it. This is illustrated in figure 5.19. We only need to specify the aspect of the world state that we're interested in (holding an object), rather than the complete state that it has previously encountered holding objects in (i.e. with the hand and object in the same location). The robot should then be able to chain together the relevant motor primitives required to achieve this higher level goal.

Target world state
Holding object 3

Figure 5.19: The iCub is given the target of holding an object.

5.6.5 Results

Upon being presented with the object and the request for the final state of the robot to be holding that object the robot constructs a chain of schemas, shown in figure 5.20, which makes use of the generalised reaching schema to satisfy the pre-conditions of the generalised grasping schema, which in turn satisfies the final target of the robot holding the object. The execution of this chain results in the robot reaching out to touch the object and then successfully grasping it.

Pre-conditions	Action	Post-conditions
Obj. $a at $x, $y	*Target:* Hand at $x, $y	Obj. $a at $x, $y Hand at $x, $y Touching obj. $a

Pre-conditions	Action	Post-conditions
Obj. $a at $x, $y Touching obj. $a	Grasp	Obj. $a at $x, $y Hand at $x, $y Holding obj. $a

Figure 5.20: The touching and grasping schemas can be chained together to form a plan of action which allows the robot to reach towards and then grasp an object at any location.

5.7 Experiment 6: Reasoning from unreliable information

In this experiment we investigate the unreliable qualities of the previously learnt pointing schema, shown in figure 5.21.

Pre-conditions	Action	Post-conditions
Unreachable obj. $a at $x, $y	*Target:* Hand at $x, $y	Reachable obj. $a at 14, 76.12 Hand at $x, $y

Figure 5.21: Generalised pointing schema learnt in earlier experiments.

This schema assumes that the object will be placed in a specific location, however in this experiment we will defy this expectation by instead moving the object to a new reachable location in which the object has not yet been encountered.

To allow for a more complex chain of actions only the iCub is used, making it possible for us to request the robot go from a position in which the object is out of reach to one in which the object is being held by the robot. This involves the construction of a longer chain of schemas than previously encountered including the use of the pointing schema. The final requested world state is illustrated in

figure 5.22.

Target world state
Holding object 3

Figure 5.22: The iCub is given the target of holding an object, starting from an object being out of reach.

5.7.1 Results

Upon being provided with the target state the robot forms a chain of actions in which the generalised pointing schema is used to request that the object be moved to the visual position 14, 76.12 in which it previously saw the object placed most frequently during the learning of the pointing schema. Once the object is in the new location the generalised reaching schema is used to touch the object and finally the grasping schema is used to grasp it. This chain is shown in figure 5.23.

To show the robot's belief in how this chain of actions will unfold we have provided the instantiated version of all the generalised schemas used during the chaining process. This is shown in figure 5.24.

Although the generalised schemas are instantiated during the planning process, these instantiated schemas are never used directly. Instead the chain of uninstantiated generalised schemas is made use of for actual execution. This means that each generalised schema can be instantiated immediately prior to execution based on the real state of the world, rather than the predicted state.

When we place the object in an unexpected location this does not interrupt the currently executing plan because the generalised touching schema is then instantiated based upon the new location. The final instantiated schemas used by the robot are shown in figure 5.25.

However, this strategy can result in potentially ambiguous conditions when multiple combinations of stimuli satisfy the requirements of a generalised schema. As such a more robust, although slightly more computationally expensive, approach would be to utilise the instantiated schemas from the planning stage and then employ the system's ability to recalculate plans when an expected condition is not met. This ability to adapt plans is demonstrated in section 5.8.

Pre-conditions	Action	Post-conditions
Unreachable obj. $a at $x, $y	*Target:* Hand at $x, $y	Reachable obj. $a at 14, 76.12 Hand at $x, $y

Pre-conditions	Action	Post-conditions
Reachable obj. $a at $x, $y	*Target:* Hand at $x, $y	Reachable obj. $a at $x, $y Hand at $x, $y Touching obj. $a

Pre-conditions	Action	Post-conditions
Reachable obj. $a at $x, $y Touching obj. $a	Grasp	Reachable obj. $a at $x, $y Hand at $x, $y Holding obj. $a
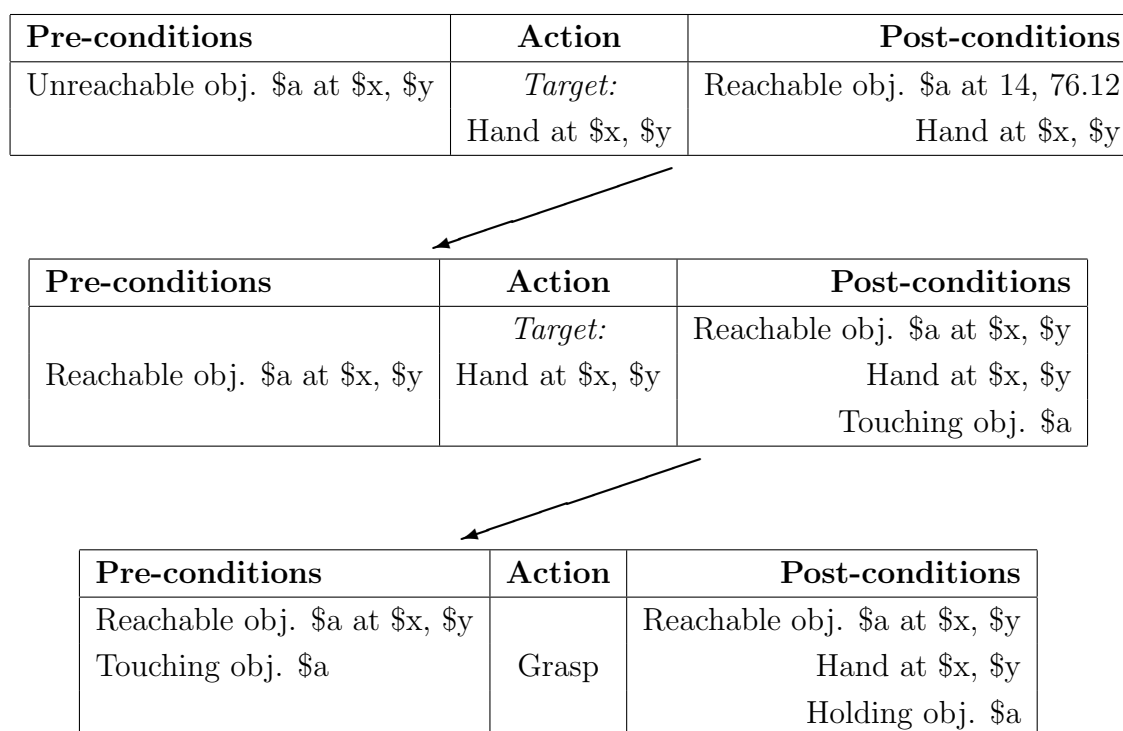
Figure 5.23: The generalised pointing, touching and grasping schemas are chained together, despite the unreliability of the pointing schema's predictions the robot is able to to request a distant object, reach out to it and then grasp it.

Pre-conditions	Action	Post-conditions
Unreachable obj. 3 at 20.38, 55.62	Reach to 20.38, 55.62	Reachable obj. 3 at 14, 76.12 Hand at 20.38, 55.62

Pre-conditions	Action	Post-conditions
Reachable obj. 3 at 14, 76.12	Reach to 14, 76.12	Reachable obj. 3 at 14, 76.12 Hand at 14, 76.12 Touching obj. 3

Pre-conditions	Action	Post-conditions
Reachable obj. 3 at 14, 76.12 Touching obj. 3	Grasp	Reachable obj. 3 at 14, 76.12 Hand at 14, 76.12 Holding obj. 3
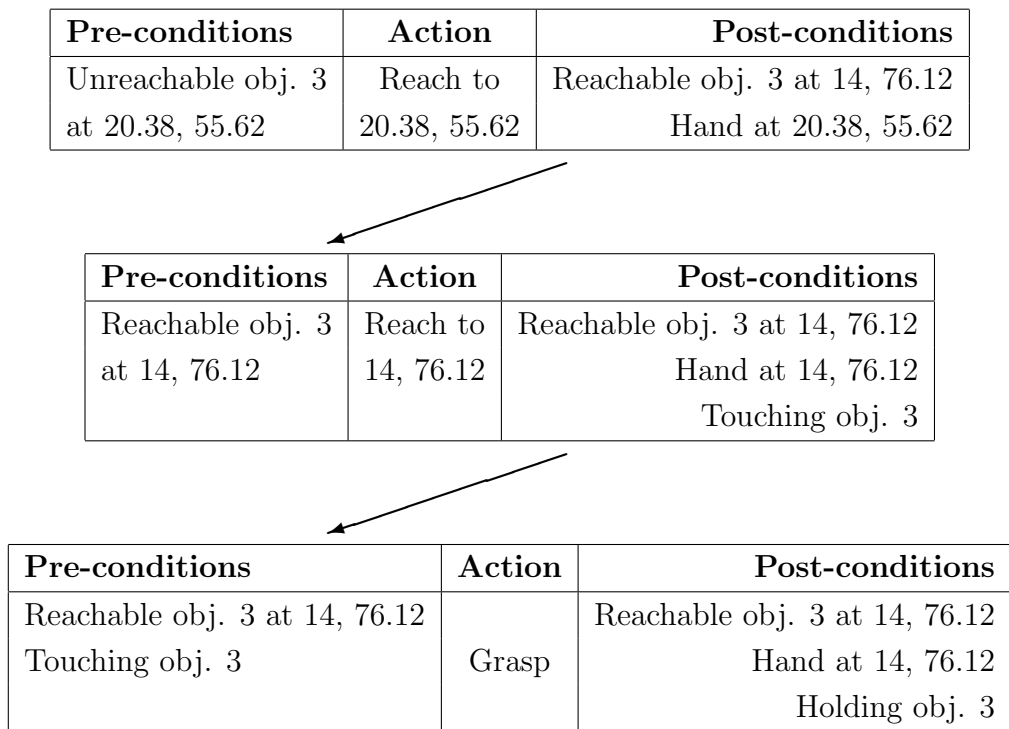
Figure 5.24: The iCub's plan for grasping an out of reach object, with all of the generalised schemas instantiated with concrete values, prior to execution beginning.

Pre-conditions	Action	Post-conditions
Unreachable obj. 3 at 20.38, 55.62	Reach to 20.38, 55.62	Reachable obj. 3 at 14, 76.12 Hand at 20.38, 55.62

Pre-conditions	Action	Post-conditions
Reachable obj. 3 at 3.4, 74.95	Reach to 3.4, 74.95	Reachable obj. 3 at 3.4, 74.95 Hand at 3.4, 74.95 Touching obj. 3

Pre-conditions	Action	Post-conditions
Reachable obj. 3 at 3.4, 74.95 Touching obj. 3	Grasp	Reachable obj. 3 at 74..95, 3.4 Hand at 3.4, 74.95 Holding obj. 3
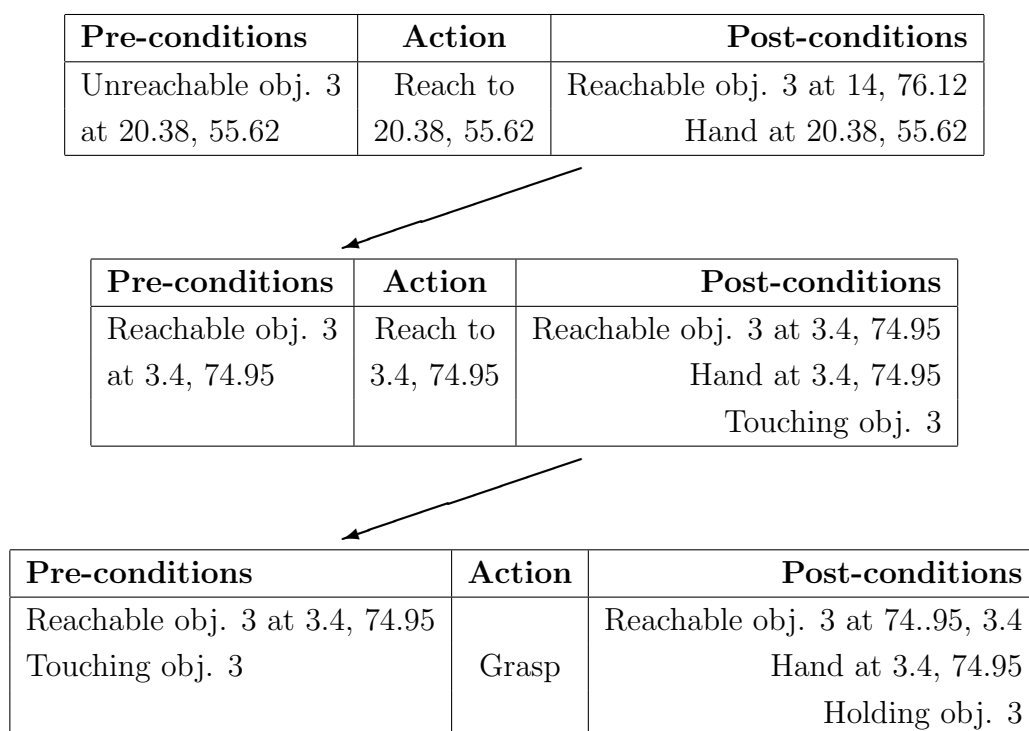
Figure 5.25: Instantiated version of the actual schemas executed when attempting to grasp an out of reach object after the object has been placed in an unexpected location.

Despite the potential for the robot's prediction of the object's final location being incorrect it is still beneficial for the robot to make this prediction when planning. By making use of the unreliable information in this schema it is able to produce a chain of schemas which can then be refined during execution to cope with the real outcome. If the robot simply ignored this information as being too unreliable it would be unable to form a chain in the first place, by making use of a combination of unreliable information when planning combined with the capacity to adapt during execution the robot is able to perform reliably in uncertain circumstances.

5.8 Experiment 7: Coping with interference from other agents

In this experiment we investigate the robot's capacity to deal with active interference in its plans. This differs from the previous experiment in that rather than simply requiring the robot to make use of information that is difficult to predict, we actively work against the robot to contradict its expectations.

We give the iCub the target state of holding an object, figure 5.26, and place an object in front of it. As the robot begins to reach towards this location we move the object to a second location, requiring the robot to reconsider its plan of action.

Target world state
Holding object 3

Figure 5.26: The iCub is given the target of holding an object.

5.8.1 Results

Upon being presented with the object and the target condition the schema memory generates the same plan of action seen in 5.6.4, connecting the generalised reaching schema together with the generalised grasping schema to allow it to reach out and grasp the object. This plan is shown in figure 5.27.

When executing a chain of schemas the robot checks the pre-conditions of each schema in the chain immediately prior to its execution. If the previous step

Pre-conditions	Action	Post-conditions
Obj. $a at $x, $y	*Target:* Hand at $x, $y	Obj. $a at $x, $y Hand at $x, $y Touching obj. $a

Pre-conditions	Action	Post-conditions
Obj. $a at $x, $y Touching obj. $a	Grasp	Obj. $a at $x, $y Hand at $x, $y Holding obj. $a
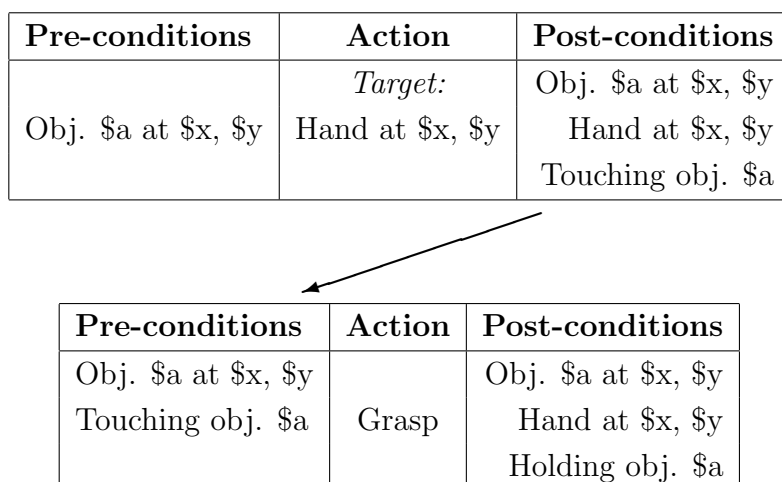
Figure 5.27: The touching and grasping schemas are chained together to form a plan prior to the interference of another agent.

in the chain has not gone according to the robot's predictions and is no longer compatible with the next step in the chain it constructs a new plan starting from the current state of the world.

Figure 5.28 shows the schemas executed by the robot, instantiated with their final values. The robot initially reaches out towards the object's first location, during this the object is moved to a new location. This is no longer compatible with the grasping schema as this requires that the robot is touching the object before it can be executed, a new plan is then arrived at which involves first reaching to the new location. After reaching to the new location the robot is able to successfully grasp the object.

The final state of the robot, the initial starting location and the location to which the object was moved can be seen in figure 5.29.

Pre-conditions	Action	Post-conditions
Obj. 3 at 19.74, 81.98	Reach to 19.74, 81.98	Obj. 3 at 19.74, 81.98
		Hand at 19.74, 81.98
		Touching obj. 3

Interference: The object is moved to a new location while the robot reaches for it. The chain is then recalculated, with the reaching step reintroduced to move the hand toward the new location:

Pre-conditions	Action	Post-conditions
Obj. 3 at 3.4, 74.95	Reach to 3.4, 74.95	Obj. 3 at 3.4, 74.95
		Hand at 3.4, 74.95
		Touching obj. 3

Pre-conditions	Action	Post-conditions
Obj. 3 at 3.4, 74.95	Grasp	Obj. 3 at 3.4, 74.95
Touching obj. 3		Hand at 3.4, 74.95
		Holding obj. 3

Figure 5.28: The final chain of action accounting for third-party interference, with all generalised schemas instantiated with concrete values.
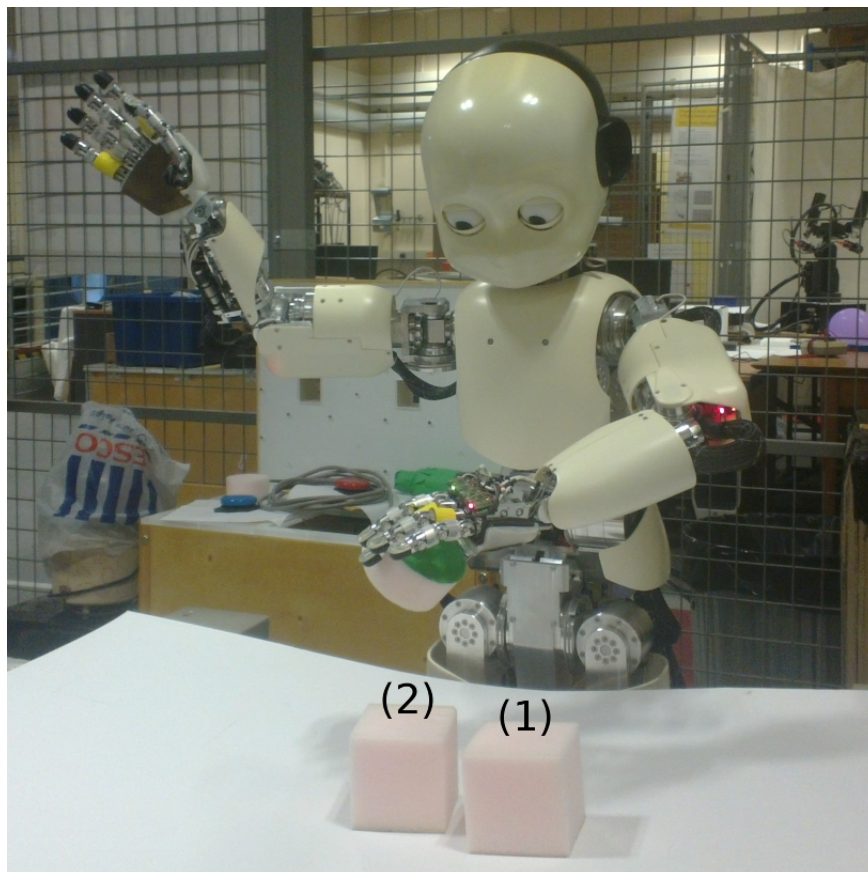
Figure 5.29: The iCub after having successfully grasped an object, despite inter-ference from a third party. The object was originally placed at position (1), then as the robot reached towards it the object was moved to position (2).

5.9 Experiment 8: Learning and responding to verbs

This experiment was carried out on the Adept system, an extended variation of this experiment can be seen on the iCub in section 5.13.

For this experiment the system starts in the end condition of experiment 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 by executing the most excited schemas in its memory, until the excitation provided by the novelty of the object drops sufficiently for it to begin executing 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.

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

5.9.1 Results

Figure 5.30 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.

Figure 5.30: *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.

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 commands. As can be seen from these results it is only necessary to give

a single example of a word for it to be potentially used as a command to direct the robot back to the action being performed at that time. Although it should be noted that while the intrinsically motivated excitation calculator is useful for guiding the robot's learning of new capabilities, it is inappropriate for commanding a robot, as the robot's interest in the spoken words gradually decrease as they become familiar. Once the excitation falls below that of the next most excited schema that other schema will be executed in favour of the schema relating to the word *'touch'*, until the excitation of that schema diminishes enough for the touching schema to once again be the most exciting. This is a product of the intrinsic motivation system being targeted towards the learning of new information, as the actions related to the word *'touch'* are not producing any new experiences the robot switches to testing out other actions in an attempt to elicit new responses from the world.

5.10 Experiment 9: Learning verbs in a noisy environment

Similar to the previous experiment we attempt to teach the robot the word *'touch'*, however this time we do so with varying levels of linguistic noise. For example rather than simply hearing the word *'touch'* in association with a touching event the robot may hear a more complete sentence such as *'this is touch'*. Similar sentences are then also used in other contexts, such as *'this is grasp'* and *'this is drop'*.

This tests the capabilities of the excitation mechanism which relate to the proportion of times an associated observation has been heard in relation to a specific act. For example the word *'touch'* is only heard in relation to touching actions, whereas the word *'this'* is heard in many different contexts. As such hearing *'touch'* will contribute a lot of excitation to touching related schemas, whereas *'this'* will only contribute a small amount of excitation to a wide range of different schemas.

5.10.1 Results

Figure 5.31 shows the excitation contributed to the selected schema for each word in the three sentences. This contribution measurement is made prior to the schema's overall excitation being diminished by either the frequency of that schema's activation or by the length of the path required to reach it.

In each case we see that the word most relevant to a specific action is given more saliency when considering the excitation of that action, while the more general words (*'this'* and *'is'*) produce a lesser contribution. This difference becomes more pronounced if the general words are associated with even more actions, figure 5.32 shows the difference in excitation contribution between the word *'this'* and the word *'grasp'* as the word *'this'* is associated with increasing numbers of different actions.
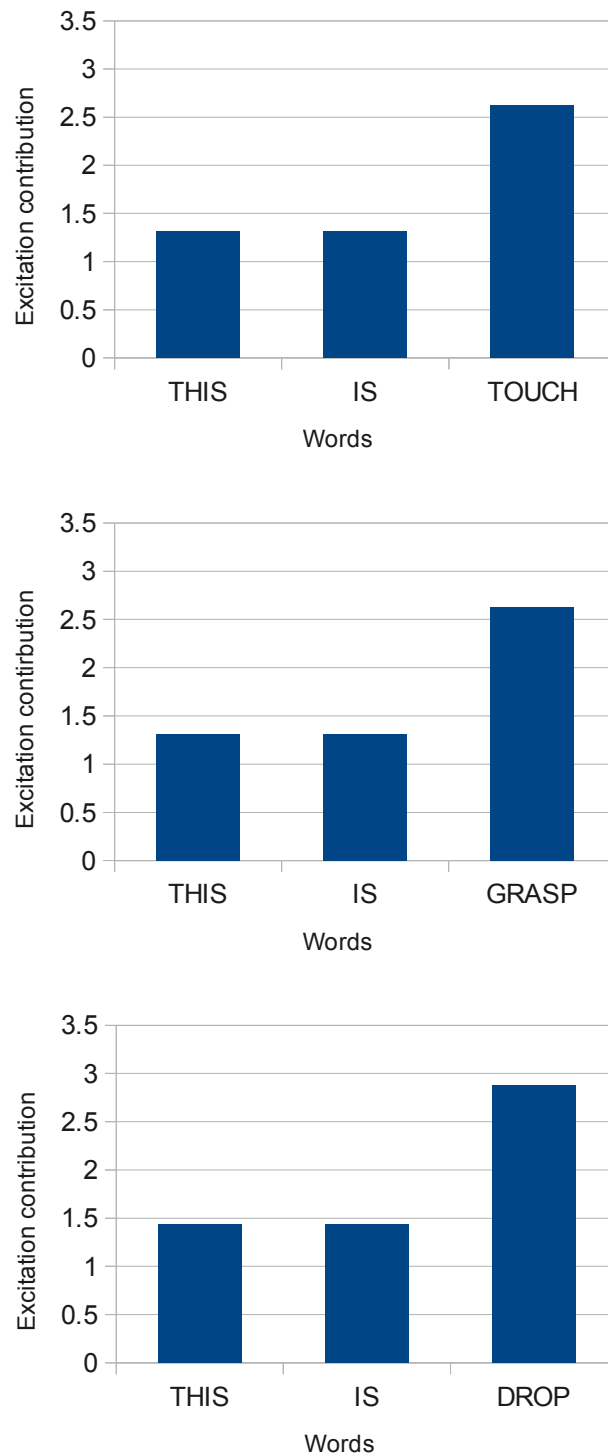
Figure 5.31: The excitation contribution that each word in a sentence makes to the schema that was eventually selected as being the most exciting.
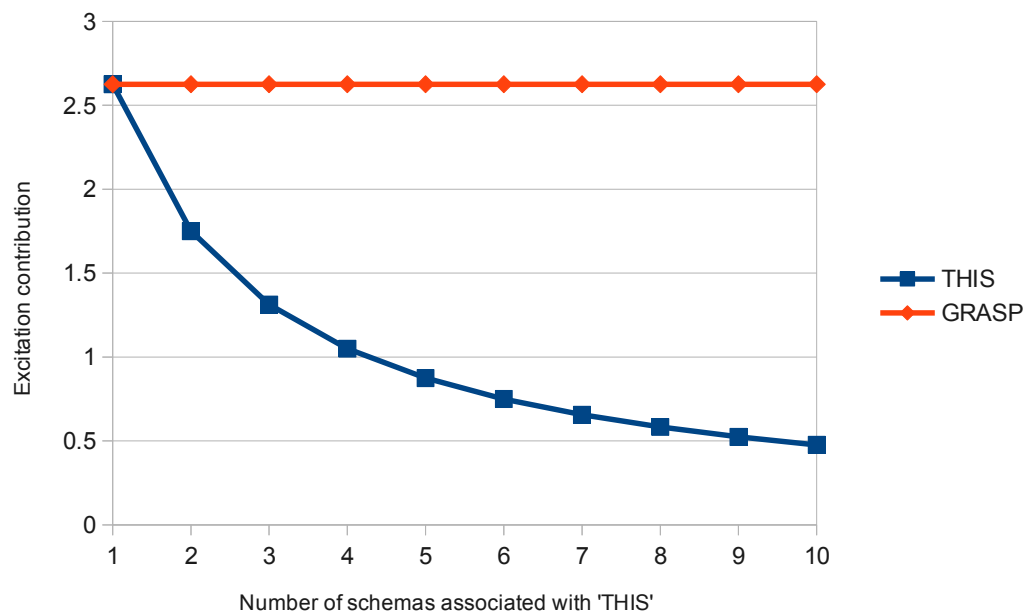
Figure 5.32: The excitation contribution towards the grasping schema of a general word, *'this'*, and an action specific word, *'grasp'*, as the general word becomes associated with increasing numbers of different schemas.

5.11 Experiment 10: Learning nouns and adjectives

The capability of the system to associate observations with other observations, discussed in section 3.2.4, includes a capacity to generalise these associations so as to focus on specific sub-components (discussed in 3.7). This means that given enough examples of otherwise different red objects alongside the word *'red'* the system should be able to form an association which accurately links the word to the correct component of these visual observations.

We effectively use colours as nouns in our experiments as the robot's visual system identifies individual objects based upon their colour.

We make use of a schema memory that has previously developed on the iCub and which has the capability to reach towards objects and grasp them with an understanding that objects that are too far away are unreachable. However, the experiment itself does not make use of the iCub, instead we simply feed stimuli to the memory and record the resulting interpretation.

We first teach the schema memory the words *'red'* and *'green'* as names for two objects, when either a red or green object is presented in a number of different locations. Each object is introduced separately from the other. We then record the associations generated. In experiment 11 we take this further and use the learned associations to assist in an object selection task.

In addition to learning the words *'red'* and *'green'* as names for specific objects, we then also teach the memory the adjectives *'near'* and *'far'*. We do this by presenting the stimuli of a number of different objects either inside the reachable space coupled with hearing the word *'near'*, or outside of the reachable space alongside the word *'far'*.

5.11.1 Results

Figures 5.33 and 5.34 show the generalised associations arrived at by the schema memory after being introduced to the red and green objects alongside their corresponding names.

The colour and ID components both retain specific values, relating to the object colour. The object ID remains specific because the underlying visual system identifies objects purely based upon their colour as discussed in chapter 4. All other components are generalised, which in the context of observation associations effectively means that they are ignored. When comparing a specific observation encountered in the world with a generalised associated observation the generalised values are populated based upon the specific observation, more detail on this process can be found in chapter 3.2.4.
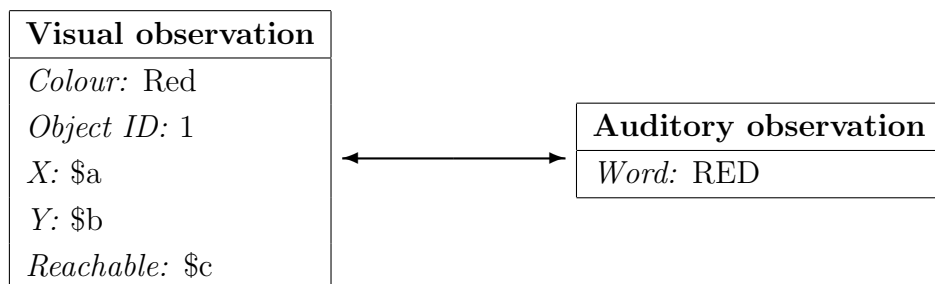
Figure 5.33: An association between a red object and the word *'red'*.

Visual observation
Colour: Green
Object ID: 3
X: $a
Y: $b
Reachable: $c

Auditory observation
Word: GREEN
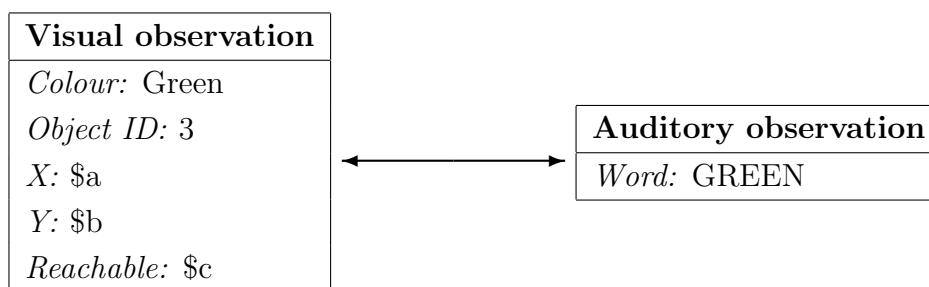
Figure 5.34: An association between a green object and the word *'green'*.

Figures 5.35 and 5.36 show the associations learnt between objects inside the robot's workspace and the word *'near'* and those outside the robot's workspace with the word *'far'*.

All components of the visual observation other than the reachability property are generalised allowing these associations to be applied to any objects in any locations dependant purely on their distance from the robot.

Visual observation
Colour: $a
Object ID: $b
X: $c
Y: $d
Reachable: True

Auditory observation
Word: NEAR
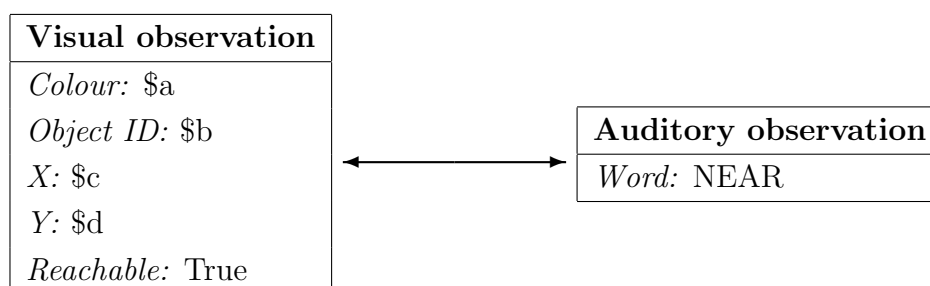
Figure 5.35: An association between an object within the robot's reach space and the word *'near'*.

Visual observation
Colour: $a
Object ID: $b
X: $c
Y: $d
Reachable: False

Auditory observation
Word: FAR
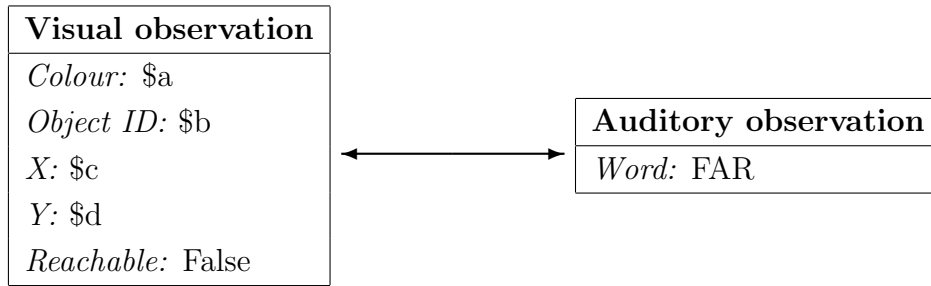
Figure 5.36: An association between an object placed outside of the robot's reach space and the word *'far'*.

5.12 Experiment 11: Responding to simple sentences

Building on the previous three experiments it should then be possible to direct the schema memory's attention to an object in a more specific way, through a simple sentence consisting of a noun-verb pair. Instead of simply touching any object when hearing the word *'touch'* or performing any action on a red object when hearing the word *'red'* we should be able to direct the schema memory's attention towards schemas which would result in the robot touching the red object with a sentence of the form *'touch red'* (because no consideration is given to word ordering this sentence could equally be presented as *'red touch'*).

5.12.1 Results

Having two objects, a red one and a green one, simultaneously placed in different locations results in the possibility for generalised schemas involving a single object to be instantiated in two different configurations. The two potential instantiations of the touching schema are illustrated in figure 5.37.

When hearing the word *'touch'* the overall excitation of both of these interpretations are equal, shown in figure 5.38.

Upon hearing just the word *'red'* the red instantiation of the grasp schema is more excited than either of the touch instantiations, due to grasping having been encountered much less frequently than touching prior to the experiment beginning. The overall excitation for both the red and green interpretations of the touch and grasp schemas are shown in figure 5.39.

Pre-conditions	Action	Post-conditions
Obj. 1 (red) at 18.86, 78.79	Hand at 18.86, 78.79	Obj. 1 (red) at 18.86, 78.79 Hand at 18.86, 78.79 Touching obj. 1 (red)

Pre-conditions	Action	Post-conditions
Obj. 3 (green) at 21.35, 75.94	Hand at 21.35, 75.94	Obj. 3 (green) at 21.35, 75.94 Hand at 21.35, 75.94 Touching obj. 3 (green)

Figure 5.37: The two possible schemas which can be generated from instantiating the generalised touching schema.
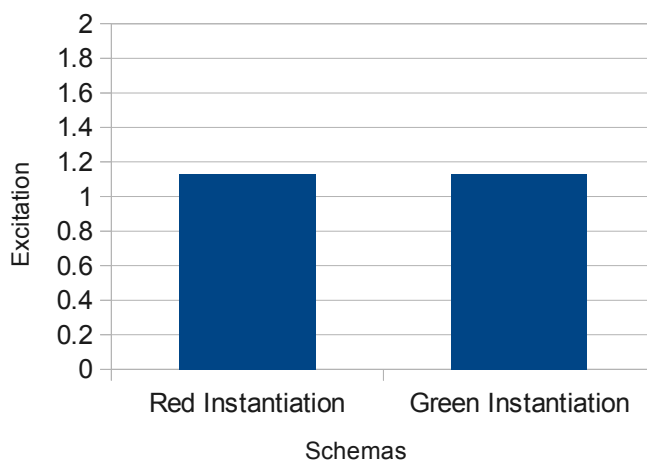
Figure 5.38: The total excitation of the two possible touch schema instantiations upon hearing the word *'touch'*.
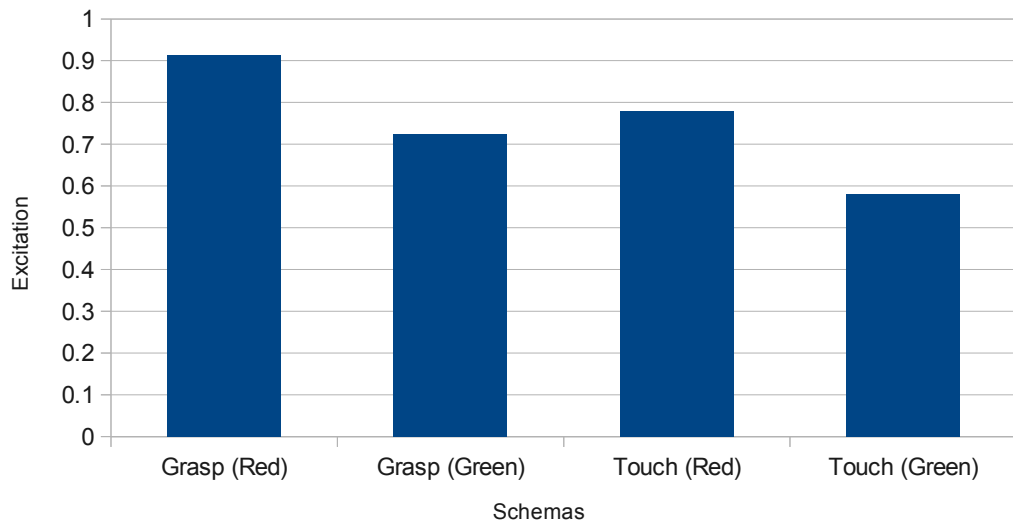
Figure 5.39: The total excitation of each possible instantiation of the touch and grasp schemas upon hearing the word *'red'*.

However, upon hearing either the phrase *'touch red'* (figure 5.40) or *'touch green'* (figure 5.41) both interpretations gain excitation overall due to there being more stimuli to cause excitation, but the instantiation of the touch schema which corresponds with the colour heard becomes more exciting than the alternative version.

If the word *'touch'* is heard either the red or green object may be touched and if just the name of an object is heard, such as *'red'*, then the red instantiation of the generalised grasp schema becomes most excited. However if the words *'touch red'* are heard then the excitation contributed from both words combine to cause the execution of the red instantiation of the generalised touching schema.
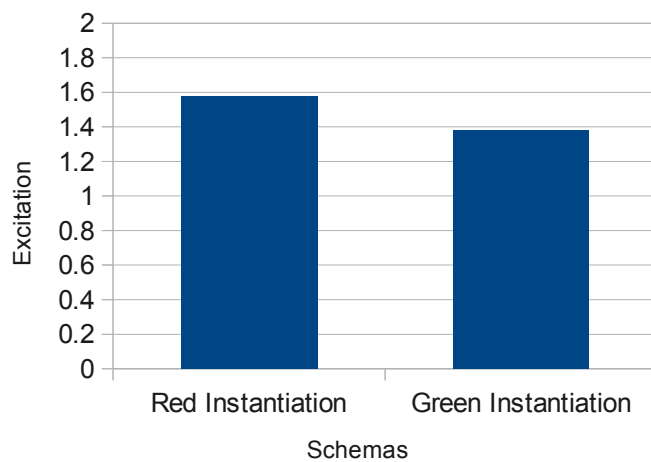
Figure 5.40: The total excitation of the two possible touch schema instantiations upon hearing the sentence *'touch red'*.
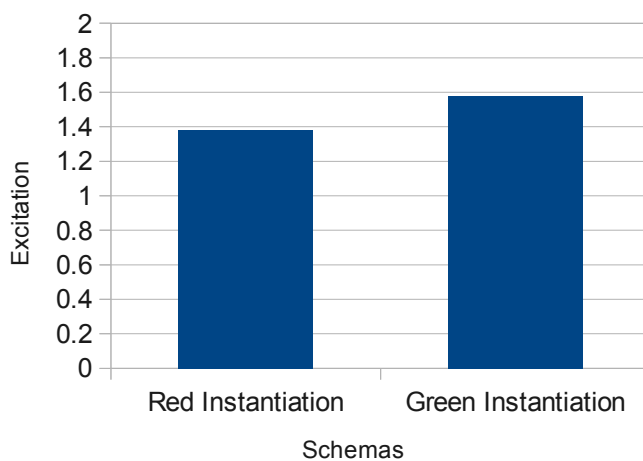
Figure 5.41: The total excitation of the two possible touch schema instantiations upon hearing the sentence *'touch green'*.

5.13 Experiment 12: Stacking objects with linguistic scaffolding

This experiment was only conducted on the iCub due to the requirement for grasping.

The robot begins with the knowledge gained from experiments 5.2 and 5.5, giving it the ability to reach towards and grasp objects. In addition to this, prior to the experimental stage beginning when the robot randomly selected a releasing action the word *'drop'* was spoken, regardless of whether or not it was currently holding an object. This dropping schema can be seen in figure 5.42, as it was learnt at a time when no objects were being held the robot has not yet found any result from performing this action.

The stack observation is a simple representation of the visual system having detected one object being on top of another. With more accurate and repeatable depth perception this relationship could instead be learnt as a change in the height of an object after being placed on other objects, rather than having explicit detection for one object being on top of another in the visual system.

Pre-conditions	Action	Post-conditions
	Release	

Figure 5.42: The initial dropping schema, generated without an object present.

The experiment begins with a single object being placed in the robot's work area in a previously unused location. The robot is then allowed to act based on its intrinsic motivation until the object has been grasped. A second object is then introduced in a new location. When the novelty of the second object attracts the robot's attention and it reaches towards it (whilst still holding the original object) we say the word *'drop'*. This should remind the robot of the releasing actions it performed previously and result in the robot constructing a small tower of objects.

5.13.1 Results

When the first object is introduced the generalised touching schema is excited, shown in figure 5.43.

Pre-conditions	Action	Post-conditions
	Target:	Obj. $a at $x, $y
Obj. $a at $x, $y	Hand at $x, $y	Hand at $x, $y
		Touching obj. $a

Figure 5.43: The generalised touching schema is excited by the presence of the first object.

This then leads on to the grasping schema becoming excited based upon the sensation of touching the object, resulting in the robot holding the first object. Once the second object is introduced the touching schema is excited again, causing the robot to reach out towards the new object, whilst still holding the original object.

At this point if the robot is left to continue acting purely based on the current stimuli it may reach to this location a number of times, however eventually the excitement caused by the new object will diminish in the absence of any unexpected results and the robot will start activating other less interesting schemas, resulting in it reaching to other locations, eventually the releasing schema will get activated but it is unlikely that the robot will be near the second object at this time. If we allow the robot to play randomly for a long period of time it might stumble upon the ability to stack objects eventually.

Instead of this random trial and error approach we make use of the language learning the robot achieved in its earlier stages of development. When the robot holds the first object over the second we say the word *'drop'*. This word became associated with the releasing schema during the robot's early development and contributes to exciting that schema now. The relative rarity of hearing auditory input makes this very exciting, and so causes the associated releasing schema to be executed.

Upon dropping the object the robot discovers that a new unpredicted visual sensation is experienced, of seeing one object stacked on top of another, and so a new schema is created to represent this, shown in figure 5.44. The final state immediately following the robot having created a small stack of objects can be seen in figure 5.45.

Pre-conditions	Action	Post-conditions
		Obj. 1 at -3.63, 80.09
Obj. 1 at -3.63, 80.09		Obj. 3 at -3.64, 80.09
Obj. 3 at -3.64, 80.09	Release	Hand at -3.64, 80.09
Holding Obj. 3		Touching obj. 1
		Touching obj. 3
		Obj. 3 stacked on Obj 1

Figure 5.44: A new schema is then created representing the stacking of one object on top of another.
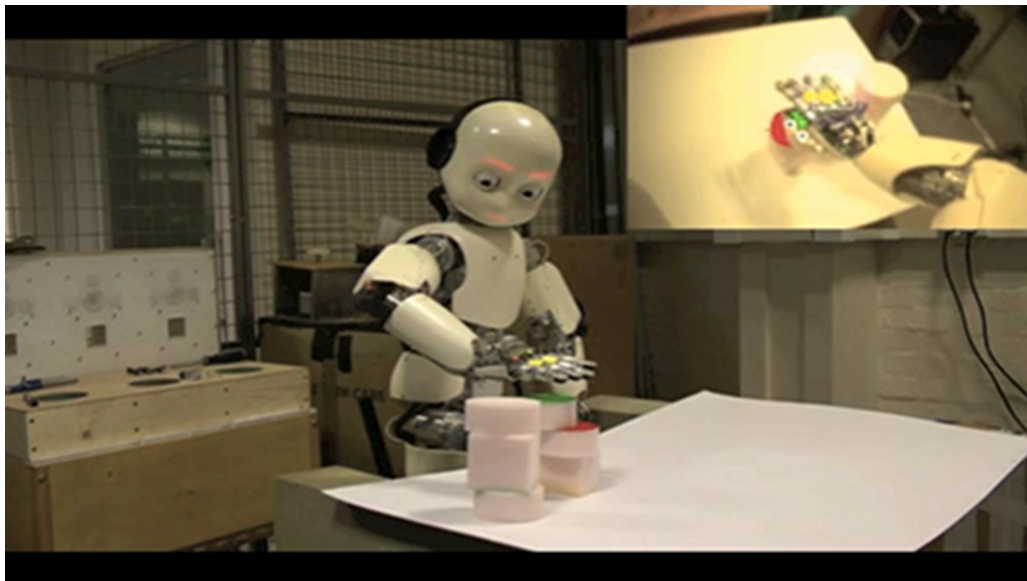
Figure 5.45: The iCub, having constructed a stack of two objects. The square object at the base of the stack is invisible to the iCub and is used to ensure that objects are a safe distance above the work surface, from the iCub's perspective it has created a stack consisting of the red object and the green object.

5.14 Experiment 13: Interpreting others' actions

In this experiment we investigate the schema framework's potential for interpreting the actions of other agents. By considering the state of the world prior to an agent acting and the state of the world after the agent has acted the schema memory is able to infer the action it would have taken in the same context to achieve that result.

After recognising another agent's action from an egocentric perspective the memory can then determine which previously useful schema chains this action might belong to. This then allows the system to make a number of guesses as to the other agent's final intention.

We make use of a memory that has previously developed on the iCub and which has the capability to reach towards objects and grasp them. However, the experiment itself does not make use of the iCub, instead we simply feed stimuli to the memory and record the resulting interpretation.

We present the memory with the sensation of first seeing an object, and then seeing another agent's hand touching the object. From this we request the memory's interpretation of what the other agent has just done and what they might be planning to do next.

5.14.1 Results

Upon being presented with information about the initial conditions of the world and the other agent's effect upon the world, but lacking any concrete information about the action performed, the schema memory is able to relate this to its own actions and determine the schema that it would have executed to achieve the same effect in that context. Figure 5.46 shows this process occurring.

While the results of the other agent's actions are given in terms of specific values the schema memory is able to relate this to a generalised representation of the act of reaching out and touching an object, giving the robot a more general understanding of the action taking place.

Pre-conditions	Action	Post-conditions
Obj. 3 at -3.24, 80.70		Obj. 3 at -3.24, 80.70
		Hand at -3.24, 80.70
		Touching obj. 3

Pre-conditions	Action	Post-conditions
	Target:	Obj. $a at $x,$y
Obj. $a at $x,$y	Hand at $x,$y	Hand at $x,$y
		Touching obj. $a
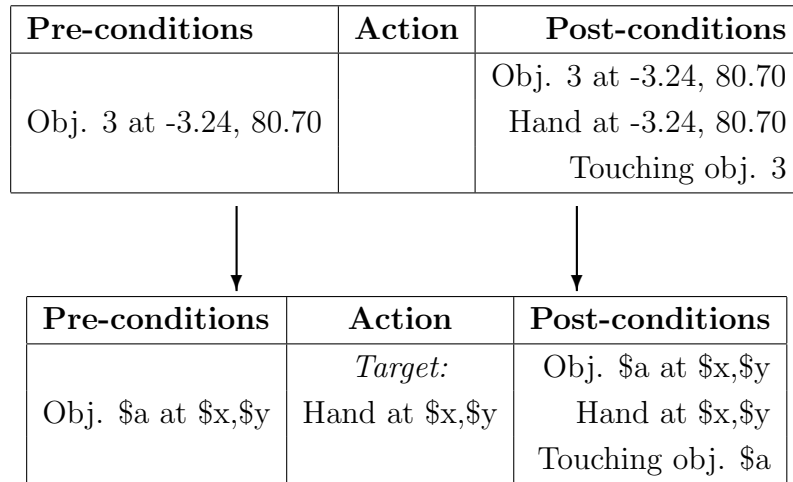
Figure 5.46: The actions of another agent are interpreted in terms of actions learnt by the iCub.

Once the other agent's actions have been related to the schema memory's own action representations it is able to consider this in the context of chains of actions which the robot previously found useful itself. In this way it is able to predict what the next action the other agent may take and what their ultimate goal may be. In this case the memory finds that the touching schema has previously been useful in a chain of actions leading to holding an object and so predicts that this may be what the other agent may be engaged in. This final prediction of the other agent's intentions is illustrated in figure 5.47.

Pre-conditions	Action	Post-conditions
Obj. $a at $x, $y	*Target:* Hand at $x, $y	Obj. $a at $x, $y Hand at $x, $y Touching obj. $a

Pre-conditions	Action	Post-conditions
Obj. $a at $x, $y Touching obj. $a	Grasp	Obj. $a at $x, $y Hand at $x, $y Holding obj. $a

Figure 5.47: The chain of potential actions inferred from observing another agent acting upon the world.

5.15 Overall progression

Tables 5.3 and 5.4 summarise the progression taken by the two robots, in the case of the Adept this includes learning to point and to touch, and in the case of the iCub this also includes grasping and forming a stack of objects. In each case we show the number of specific schemas learnt, the number of generalised schemas learnt and the number of actions taken to achieve that learning. We believe the number of actions taken to be a more reliable measure of learning efficiency than the actual time taken, as the time is largely dependent upon factors such as the robot's motor speed and in the case of certain experiments is influenced by the time taken for other agents to provide social feedback.

Scenario	Schemas learnt	Generalised schemas	Actions
Initial learning	111	0	111
Learning to touch	113	1	113
Learning to point	135	1	135

Table 5.3: Cumulative totals of schemas produced when following a progression from a 'new born' state to pointing behaviour on the Adept arm system. The 'schemas learnt' column does not include generalised schemas.

As we saw in section 5.3 the representation of space provided to the Adept

robot prevents it from being able to form a generalised schema to represent pointing, instead to construct a representation of locations in which objects can be pointed towards rather than touched it must explicitly experience the act of pointing towards these locations and form a specific schema for each one. With the size of the visual fields and valid joint configurations of the robot this results in twenty two fields in which the robot can point towards objects, each of which must be learnt individually. Whereas on the iCub only two examples of distant objects are required, after which a generalised schema can be created.

Scenario	Schemas learnt	Generalised schemas	Actions
Initial learning	50	0	50
Learning to touch	52	1	57
Learning to point	54	2	59
Learning to grasp	56	3	63
Stacking objects	57	3	67

Table 5.4: Cumulative totals of schemas produced when following a progression from a 'new born' state to object stacking on the iCub. The 'schemas learnt' column does not include generalised schemas.

A disparity can be observed in the number of actions taken when learning the generalised touching schema versus the grasping or pointing schemas. This is because the bias towards repetition of new schemas causes the robot to repeat the touching action a number of times, however in the case of grasping and pointing the conditions under which these schemas can be executed are no longer present in the environment, so while these schemas retain a high level of excitation themselves the fact that a chain of schemas is required to activate them reduces their overall excitation in the current environmental conditions. Were these conditions to be repeated, e.g. if the robot were to be presented with a distant object to point towards, then these schemas would be excited and result in the repetition of that action.

We believe that the low number of actions required to learn useful general representations of different skills makes this approach particularly suited to social learning with human participants. This allows the human social partners to interact in a relatively natural way with the robot, as it requires only a small number of examples to form workable hypotheses.

Chapter 6

Conclusions

In this thesis we have presented a framework for schema learning containing a novel algorithm for schema generalisation through parameterisation, and an approach to associated experiences that allow for language learning in an otherwise causally oriented framework.

We have then made use of this schema learning mechanism to support the investigation of a psychologically inspired developmental progression and have demonstrated how this progression can simplify certain learning tasks.

To drive action in this framework we have developed an intrinsic motivation algorithm suitable for directing the schema learning system towards actions likely to elicit further knowledge about the environment.

These advances have all been combined into a series of experiments culminating in a humanoid robot starting with very little knowledge about the world first learning how to interact with single objects by touching and then grasping them and eventually developing the skills necessary to build a stack of objects and interpret the actions of other agents.

We demonstrated three different ways in which the robot's decision making could take place. First, through purely intrinsically motivated excitation based around a combination of the novel aspects of the world and remembered experiences. Second, by providing the robot with an explicit goal and allowing it to produce a plan of action utilising the knowledge gained during its earlier intrinsically motivated learning. Third, by using language to influence the robot's internal motivation and so guide it towards actions which the robot might not

otherwise be reminded of by the current state of the world. This third approach extends the first to include social influences in the robot's motivation.

6.1 Contributions

Below we discuss the four main contributions made by this work, summarising the findings of the experiments performed upon the system and highlighting the interdependence of these contributions, with each providing useful support and extensions to other aspects of the system.

6.1.1 Developmental progression

Throughout the experiments in chapter 5 we have employed a staged, developmental process, with each experiment building upon the competencies developed during earlier experiments.

Initially the robot performs some basic motor babbling and learns the results of its most basic movements, after which we introduced objects for the robot to interact with. From this it develops general representations for touching and grasping objects, requiring only a small number of examples.

A comparison was then performed of how two robotic systems may arrive at different representations for pointing gestures, with these pointing gestures arising out of more general reaching behaviour. This remains consistent with the idea that proto-imperative pointing may develop as a result of early reaching behaviour in infants.

While this form of pointing offers the robot a simplistic mechanism for communicating its desires to other agents, we further develop the robot's communicative capacity by introducing spoken language. First we introduce verbs which the robot learns to associate with corresponding schemas. Because the robot has been able to form generalised schemas representing entire concepts with a single schema it is possible for these verbs to be learnt quickly and associated with a whole class of actions. Employing this generalisation mechanism on associations between individual observations within a schema then allows us to teach the robot nouns and adjectives which become associated with the fine grained details within a schema. Combining these two stages then allows us to direct the system to perform specific actions on specific objects.

We then demonstrate how this learnt capacity for language can assist in further motor learning activities. A parent may try to scaffold their child's learning by directing their attention towards actions that may expose the child to new experiences. We take a similar approach by presenting the robot with two objects to play with and then once the robot is holding one object over the other we say the word *'drop'*, reminding the robot of dropping actions it has performed in the past. By dropping one object on top of another a small stack is formed, allowing the robot to experience a new way in which objects can interact with each other. The robot can then use the knowledge gained from this scaffolded example to construct stacks of objects without assistance in the future.

Finally we show how a robot can make use of its own experience to interpret the actions of other agents and their potential goals. When the system is presented with the experience of seeing another agent reach towards an object it is able to relate this to its own general concept of reaching for objects, from there it can infer that the other agent may be attempting to grasp the object as this is what the robot has used its reaching ability as a preliminary step for in the past. By having experienced these chains of actions during the robot's development it is able to understand the actions and goals of others, relating them closely to its own experience.

6.1.2 Algorithm for schema generalisation through parameterisation

We introduced a novel algorithm for schema generalisation based around the parameterisation of properties within schemas. We demonstrated how this approach allows for the fast generation of hypotheses about the world based on very sparse data, these hypotheses can then be tested by the robot applying them to new scenarios and determining their reliability.

Rather than simply determining which aspects of the schema may be interchangeable with other values as many existing schema systems do, the generalisation mechanism presented in this work attempts to find generalisable relationships between the pre-conditions, the action and the post-conditions of a schema. This allows for the formation of hypotheses about how an interaction may work at a more abstract level.

Schemas generalised in this manner allow the robot to represent entire concepts such as touching and grasping objects with a single schema. These abstract schemas are then transformed into concrete schemas based upon the current perception of the world at the time they are activated. By creating an abstract representation for a concept it becomes possible to associate words with these schemas using only a small number of examples, were these concepts represented by a large number of separate concrete schemas it would be problematic to form these associations with all the relevant schemas.

This approach also greatly reduces both the total number of schemas required to represent the results of the robot's actions and the number of actions required to learn a concept. By keeping the total number of schemas low we reduce the computational requirements of operations such as calculating schema excitation, which is performed upon all schemas, and the finding of schema chains for achieving multi-part actions.

While other generalisation mechanisms would also reduce the number of schemas required, we believe that the generalisations arrived at by our approach offer a strong benefit in terms of understandability. It is relatively easy to immediately evaluate the hypotheses generated by the robot and to understand the relationships between actions and effects that they represent.

6.1.3   Associated observations in schema learning

While the definition and use of schemas in psychology is quite broad, computational schema learning systems typically focus upon causal relationships between action and effect. This same approach forms the central mechanism within our own schema learning system, however in addition to this we introduce a capacity for associating transient experiences which occur during an action but have no permanent observable effect upon the world.

It is this facility which provides the robot with the ability to learn language. Spoken words can become associated with entire schemas, providing a framework suitable for the learning of verbs. Nouns and adjectives can then also become associated with properties within a schema, by employing the same parameterisation based generalisation mechanism used for generalising schemas, these associations

can be tailored to either individual or groups of properties whilst discarding aspects of the related sensory information that are irrelevant to the word being learnt.

These verbs can then be used to suggest specific actions, increasing the excitement of their related schemas. Typically this will involve the selection of a generalised schema which has the capacity for being instantiated in a number of different ways, for example a grasping schema may be used to grasp any of a number of objects presented to the robot. By combining the verbs with either nouns or adjectives into simple sentences a specific interpretation of a generalised schema can be given greater precedence within the excitation system.

6.1.4 Intrinsically motivated schema learning

The intrinsic motivation system introduced in this work provides the main driving force behind action within our framework. The approach taken focuses on performing actions which are likely to elicit additional information about the world and how the robot can interact with it. This is done by directing the robot's attention towards novel aspects of the environment and applying actions which appear to have some commonality with the new sensations.

Throughout the developmental progression encountered in the experiments the intrinsic motivation system was vital in selecting relevant actions which allowed the robot to quickly learn to represent the ways in which objects and other agents can be interacted with. With the addition of the language capacity provided by the associated observations it becomes possible to influence this motivation using spoken language and so guide the robot towards actions which may not otherwise be afforded by the environment. This allows for a natural scaffolding process to take place between humans and the robot.

The knowledge gained through the play behaviour emerging out of the intrinsic motivation system can be used in the formation of chains of actions aimed at achieving a given target situation, in this way the result of the learning from the intrinsically motivated behaviour can be utilised by other motivational drives. For example, the robot could be motivated to always carry out spoken requests, moving the activation of language related schemas from being purely a suggestion aimed at assisting the robot's learning into a command with the purpose of aiding a human in a task.

6.2 Further Work

6.2.1 Discarding unnecessary pre-conditions through generalisation

The generalisation algorithm proposed in chapter 3.6 only considers the properties of a schema's observations as targets for generalisation. While this is sufficient for all of the scenarios encountered by the robot in our earlier experiments, when learning certain new skills in more complicated environments it would be beneficial for the generalisation mechanism to be able to discard entire observations from a schema's pre-conditions to increase that schema's generality.

For example if we consider a scenario in which the robot is presented with a long object with two visually distinct ends and the robot grasps the object from one end, the other end will appear in a different field from the one grasped. This then opens up scenarios in which the robot can make use of the long object as a tool to manipulate other distant objects. However before this can be achieved the robot must be given the opportunity to learn how the distant end of the object moves in relation to the grasped end. This requires the robot to go through a process similar to the original hand fixation in which it moves the object to many different locations and learns the result, however in the current implementation this would lead to the robot incorrectly assuming that the position the object was in prior to movement was a pre-condition of the schema generated to represent that movement. In the hand fixation stage this is solved by explicitly ignoring the hand as a possible source of pre-conditions, while this may be a reasonable assumption for the hand it would not be possible to make this assumption for arbitrary objects. Instead it would be more beneficial if after seeing a few examples of the object motion from different starting locations the generaliser was able to propose a new schema with the unnecessary pre-condition of the object's prior location removed.

Algorithm 9 outlines an extended version of the generalisation algorithm which includes this additional functionality. While this algorithm has been implemented and tested in a number of small test cases it was not used for the experiments detailed in this thesis, having been developed after the majority of these experiments were completed. As such the algorithm has been included here in the further work section due to the requirement for further comprehensive testing to

be performed. In addition to this the removal threshold referenced in the algorithm is currently a static threshold given a value of 2 in our tests, more work must be undertaken to determine a suitable method for automatically calculating this threshold.

Algorithm 9 Extended generalisation algorithm with the capacity for further generalisation through the removal of extraneous pre-conditions.

> **function** GENERALISE($newSchema, existingSchemas$)
>> *// Find similar schemas and possible target action*
>> $minSize$ = number of pre-conditions in $newSchema$
>> $targetUsed$ = true
>> $action$ = a new target action
>> **for** each $schema$ in $existingSchemas$ **do**
>>> **if** $schema$ already generalised **then**
>>>> **continue**
>>>
>>> **end if**
>>> **if** similar($newSchema$, $schema$) **then**
>>>> Add $schema$ to list of $similarSchemas$
>>>
>>> **end if**
>>> **if** $newSchema$'s action has no parameters **then**
>>>> $action$ = a copy of $newSchema$'s action
>>>> $targetUsed$ = false
>>>
>>> **else if** $newSchema$'s action is the same as $schema$'s action **then**
>>>> *// Find the result of this action in the simplest context*
>>>> **if** number of pre-conditions in $schema$ less than $minSize$ **and** $schema$ has predictions **and** satisfies($schema$, $newSchema$) **then**
>>>>> $minSize$ = number of pre-conditions in $schema$
>>>>> $action$'s target = $schema$'s post-conditions
>>>>
>>>> **end if**
>>>
>>> **end if**
>>
>> **end for**
>> **if** number of schemas in $similarSchemas$ less than $evidenceThreshold$ **then**
>>> *// Not enough evidence to generalise this schema yet*
>>> **return**
>>
>> **end if**
>> **if** $targetUsed$ **and** $action$'s target is null **then**
>>> *// Unable to find suitable target action*
>>> **return**
>>
>> **end if**

Algorithm 9 Extended generalisation algorithm with the capacity for further generalisation through the removal of extraneous pre-conditions (continued).

---

    for each *schema* in *similarSchemas* **do**

        // Find different generalisations possible from each similar schema

        variables = findVariables(*schema*)

        for each *observation* in *newSchema*'s pre-conditions **do**

            // Determine if pre-conditions can be removed to make a more general specific schema.

            if *observation* contains no generalised properties **then**

                for each *testSchema* in *similarSchemas* **do**

                    if *testSchema*'s post-conditions equal *newSchema*'s post-conditions **and** *testSchema*'s pre-conditions have a corresponding observation with different property values to *observation*'s **then**

                        Add *testSchema* to *removeList*

                    end if

                end for

                if Size of *removeList* greater than *removeThreshold* **then**

                    // Create new schemas with observation removed based upon newSchema and the matching testSchemas

                    simplerSchema = copy of *newSchema*

                    Remove *observation* from *simplerSchema*'s pre-conditions

                    if *simplerSchema* doesn't already exist in the schema memory **then**

                        Add *simplerSchema* to schema memory

                    end if

                    for each *removeSchema* in *removeList* **do**

                      simplerSchema = copy of *removeSchema*

                      Remove *observation* from *simplerSchema*'s pre-conditions

                      if *simplerSchema* doesn't already exist in the schema memory **then**

                        Add *simplerSchema* to schema memory

                      end if

                  end for

                end if

            end if

        end for

Algorithm 9 Extended generalisation algorithm with the capacity for further generalisation through the removal of extraneous pre-conditions (continued).

---

   if *variables* is empty **then**

     return

   end if

   trialSchema = copy of *newSchema*

   trialSchema's action = copy of *action*

   usedVariables = generaliseState(reference to *trialSchema*'s pre-conditions, *variables*)

   if *targetUsed* **then**

     usedVariables = *usedVariables* + generaliseState(reference to *trialSchema*'s target action, *variables*)

   end if

   generaliseState(reference to *trialSchema*'s post-conditions, *usedVariables*)

   if generalisation matching *trialSchema* already exists **then**

     continue

   end if

   satisfied = 0

   for each *testSchema* in *similarSchemas* **do**

     if satisfies(*testSchema*, *trialSchema*) **then**

       increment *satisfied*

     end if

   end for

   if (*satisfied* / number of schemas in *similarSchemas*) less than *satisfactionThreshold* **then**

     continue

   end if

   Add *testSchema* to the schema memory.

  end for

 end function

6.2.2 Extension of schema parameterisation to include conditional statements

There are cases where it could be beneficial for the generalisation mechanism discussed in 3.6 to be able to place conditions upon the parameters it devises. For example there may be situations in which it is useful to restrict a generalisation to only being applicable within one section of the visual space. This can be done by placing certain conditions upon the parameters discovered during generalisation. Figure 6.1 shows an example of how this may be represented via an additional pre-condition within a generalised schema.

Pre-conditions	Action	Post-conditions
		Object \$a at \$j, \$k
Object \$a at \$j, \$k	Hand at \$j, \$k	Hand at \$j, \$k
Condition: \$j > 20		Touching object \$a

Figure 6.1: An example of a generalised schema in which a parameter is constrained by an additional condition statement.

Currently the schema framework has the capacity to represent a schema such as this, however little work has been undertaken to allow the generalisation component to actually produce schemas of this form.

6.2.3 Constrained working memory

In the early stages of our current implementation we introduce artificial constraints on the robot's abilities, preventing it from hearing words or focusing on any objects other than its own hand. In the later stages we then constrain the complexity of the environment, gradually introducing first one, then multiple objects for it to interact with.

An alternative and potentially more flexible approach could be to constrain the robot's working memory so that it is only able to focus on a limited number of sensory stimuli at once, then gradually increase this limit as the robot becomes habituated to each stage of learning. This could be seen as similar in approach to the focus of attention studies performed by Foner and Maes, who applied restrictions to the sensory input of a schema system to reduce the complexity of the learning problem [34].

This would perhaps be likely to result in a number of additional less useful schemas, where the robot has performed actions in which it does not have enough information to accurately represent their results or their conditions in its current state but it has the advantage that the robot can be started in a more complex non-scaffolded environment.

It may also be beneficial to make the working memory limit a dynamic threshold that adjusts based on how well the robot is performing in the current environment. In this manner if a robot trained in one environment is exposed to an entirely new environment it may retreat back into an earlier stage of development, reducing its working memory, to allow it to learn the more basic elements of that environment. Once these basic elements are well understood the threshold would rise again allowing the robot to build upon these basic building blocks of understanding to better interpret this new environment.

6.2.4 Embodied translation

Due to the manner in which words are associated with sensorimotor experiences through associated observations it should be possible for the system to learn a second language without further modification to the learning framework. While work has previously looked at imbuing robots with an ability to translate between multiple languages we believe our approach could provide one of the first examples of truly embodied translation.

Due to the tight coupling of language learning to sensorimotor experiences the system would be able to make more informed judgements about translations by factoring in its current experiences of the world. For example if the robot has been trained in both Welsh and English and is asked what the name of the fruit in front of it is in Welsh it would be able to respond with the specific name of the fruit, rather than a literal translation of the word *'fruit'*.

6.2.5 Training supervised classifiers based on schema knowledge

In chapter 5.14 we demonstrated the capacity for the schema framework to recognise others' actions in terms of the concepts it had already learnt when interacting with the world itself.

By recording low level sensor data coincidental with this activity we believe it may be possible to then train a more traditional supervised classifier to recognise the action being taken by the other agent in the absence of known starting and end conditions, allowing this action to be recognised in new contexts which the robot has not previously experienced.

For example, by recording the raw visual data relating to a human hand interacting with an object and then having the schema memory tag this with the appropriate egocentric schema we begin to construct a dataset of labelled frames which can be used for training a lower level system. This system's classifications could then be fed back to the schema memory upon detection of this action in the future, providing the schema system with an additional mechanism for observing the actions of others.

6. CONCLUSIONS

Bibliography

[1] M. Arbib. *The metaphorical brain: an introduction to cybernetics as artificial intelligence and brain theory.* John Wiley & Sons, 1972. 46

[2] M. Arbib. *The metaphorical brain 2: Neural networks and beyond.* John Wiley & Sons, Inc., 1989. 41, 46

[3] M. Arbib. The mirror system, imitation, and the evolution of language. *Imitation in Animals and Artifacts*, pages 229–280, 2002. 54

[4] M. Asada, K. Hosoda, Y. Kuniyoshi, H. Ishiguro, T. Inui, Y. Yoshikawa, M. Ogino, and C. Yoshida. Cognitive developmental robotics: a survey. *Autonomous Mental Development, IEEE Transactions on*, 1(1):12–34, 2009. 36

[5] M. Asada, K. F. MacDorman, H. Ishiguro, and Y. Kuniyoshi. Cognitive developmental robotics as a new paradigm for the design of humanoid robots. *Robotics and Autonomous Systems*, 37(2):185–193, 2001. 36

[6] M. T. Banich, M. P. Milham, R. A. Atchley, N. J. Cohen, A. Webb, T. Wszalek, A. F. Kramer, Z. P. Liang, V. Barad, D. Gullett, C. Shah, and C. Brown. Prefrontal regions play a predominant role in imposing an attentional set: evidence from fMRI. *Cognitive Brain Research*, 10(1-2):1–9, 2000. 52

[7] F. C. Bartlett. *Remembering: A study in experimental and social psychology.* Cambridge University Press, 1932. 45

[8] A. G. Barto, S. Singh, and N. Chentanez. Intrinsically motivated learning of hierarchical collections of skills. In *Proceedings of the 3rd International*

Conference on Development and Learning, pages 112–19, San Diego, USA, 2004. 37

[9] J. D. Becker. A model for the encoding of experimental information. In R. C. Schank and K. M. Colby, editors, *Computer models of thought and language*, pages 396–434. W. H. Freeman and Company, San Francisco, USA, 1973. 46

[10] R. S. Blumenfeld and C. Ranganath. Dorsolateral prefrontal cortex promotes long-term memory formation through its role in working memory organization. *The Journal of Neuroscience*, 26(3):916–925, 2006. 52

[11] R. S. Blumenfeld and C. Ranganath. Prefrontal cortex and long-term memory encoding: an integrative review of findings from neuropsychology and neuroimaging. *The Neuroscientist*, 13(3):280–291, 2007. 52

[12] J. Bonaiuto, E. Rosta, and M. Arbib. Extending the mirror neuron system model, I. *Biological Cybernetics*, 96(1):9–38, 2007. 55

[13] A. H. Bond and D. H. Mott. Learning of sensory-motor schemas in a mobile robot. In *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, pages 159–161, 1981. 46

[14] G. Buccino, F. Lui, N. Canessa, I. Patteri, G. Lagravinese, F. Benuzzi, C. A. Porro, and G. Rizzolatti. Neural circuits involved in the recognition of actions performed by nonconspecifics: An fMRI study. *Journal of Cognitive Neuroscience*, 16(1):114–126, 2004. 54

[15] R. L. Buckner, W. Koutstaal, D. L. Schacter, A. D. Wagner, and B. R. Rosen. Functional-Anatomic Study of Episodic Retrieval Using fMRI: I. Retrieval Effort versus Retrieval Success. *Neuroimage*, 7(3):151–162, 1998. 52

[16] C. A. Buneo, M. R. Jarvis, A. P. Batista, and R. A. Andersen. Direct visuomotor transformations for reaching. *Nature*, 416(6881):632–636, 2002. 53

[17] S. Carey. The child as word learner. In M. Halle, J. Bresnan, and G. A. Miller, editors, *Linguistic theory and psychological reality*, pages 264–93. MIT Press, 1978. 43

[18] H. H. Chaput. *The constructivist learning architecture: a model of cognitive development for robust autonomous robots*. Computer Science Department, University of Texas at Austin, 2004. 47

[19] H. H. Chaput, B. Kuipers, and R. Miikkulainen. Constructivist learning: A neural implementation of the schema mechanism. In *Proceedings of the Workshop on Self-Organizing Maps (WSOM03)*, 2003. 47

[20] N. Chomsky. Recent contributions to the theory of innate ideas. *Synthese*, 17(1):2–11, 1967. 41

[21] R. K. Clifton, D. W. Muir, D. H. Ashmead, and M. G. Clarkson. Is visually guided reaching in early infancy a myth? *Child development*, 64(4):1099–1110, 1993. 39

[22] R. P. Cooper and D. W. Glasspool. Learning action affordances and action schemas. In *Connectionist models of learning, development and evolution: Proceedings of the Sixth Neural Computation and Psychology Workshop*, pages 133–142, 2001. 48

[23] R. P. Cooper and T. Shallice. Contention scheduling and the control of routine activities. *Cognitive Neuropsychology*, 17(4):297–338, 2000. 47

[24] R. P. Cooper and T. Shallice. Hierarchical schemas and goals in the control of sequential behavior. *Psychological Review*, 113(4):887–916, 2006. 48

[25] A. Dagher, A. M. Owen, H. Boecker, and D. J. Brooks. Mapping the network for planning: a correlational PET activation study with the Tower of London task. *Brain*, 122(10):1973–1987, 1999. 52

[26] K. Dautenhahn and A. Billard. Studying robot social cognition within a developmental psychology framework. In *In Proceedings of the 3rd European Workshop on Advanced Mobile Robots*, pages 187–194. IEEE, 1999. 41

[27] D. C. Dennett. *Consciousness explained*. Penguin, 1993. 50

[28] D. C. Dennett. Darwin's "strange inversion of reasoning". *Proceedings of the National Academy of Sciences*, 106(Supplement 1):10061–10065, 2009. 50

[29] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271, 1959. 58

[30] G. L. Drescher. *Made-Up Minds: A Constructivist Approach to Artificial Intelligence.* MIT Press, 1991. 46

[31] K. Ejiri and N. Masataka. Co-occurences of preverbal vocal behavior and motor action in early infancy. *Developmental Science*, 4(1):40–48, 2001. 39

[32] T. Erez and W. D. Smart. What does shaping mean for computational reinforcement learning? In *Proceedings of the 7th IEEE International Conference on Development and Learning*, pages 215–219, Monterey, USA, 2008. IEEE. 35

[33] R. E. Fikes and N. J. Nilsson. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial intelligence*, 2(3):189–208, 1972. 48

[34] L. Foner and P. Maes. Paying attention to what's important: Using focus of attention to improve unsupervised learning. In *Proceedings of the Third International Conference on Simulation of Adaptive Behavior*, pages 256–265, 1994. 47, 155

[35] T. Fong, I. Nourbakhsh, and K. Dautenhahn. A survey of socially interactive robots. *Robotics and Autonomous Systems*, 42(3-4):143–166, 2003. 42

[36] S. Funahashi, C. J. Bruce, and P. S. Goldman-Rakic. Dorsolateral prefrontal lesions and oculomotor delayed-response performance: evidence for mnemonic "scotomas". *The Journal of Neuroscience*, 13(4):1479–1497, 1993. 52

[37] V. Gallese, L. Fadiga, L. Fogassi, and G. Rizzolatti. Action recognition in the premotor cortex. *Brain*, 119(2):593, 1996. 54

[38] C. Geib, K. Mourao, R. Petrick, N. Pugeault, M. Steedman, N. Krueger, and F. Wörgötter. Object action complexes as an interface for planning and robot control. In *IEEE RAS International Conference on Humanoid Robots*, 2006. 48

[39] O. L. Georgeon and F. E. Ritter. An intrinsically-motivated schema mechanism to model and simulate emergent cognition. *Cognitive Systems Research*, 2011. 50

[40] O. L. Georgeon and I. Sakellariou. Designing environment-agnostic agents. In *Proceedings of ALA2012, Adaptive Learning Agents workshop at AA-MAS2012, 11th International Conference on Autonomous Agents and Multiagent Systems*, pages 25–32, Valencia, Spain, 2012. 50

[41] J. J. Gibson. *The ecological approach to visual perception*. Lawrence Erlbaum, 1986. 49

[42] H. Ginsburg and S. Opper. *Piaget's theory of intellectual development, second edition*. Prentice-Hall, 1979. 33

[43] A. M. Graybiel. The basal ganglia and chunking of action repertoires. *Neurobiology of learning and memory*, 70(1-2):119–136, 1998. 52

[44] C. Grefkes, A. Ritzl, K. Zilles, and G. R. Fink. Human medial intraparietal cortex subserves visuomotor coordinate transformation. *Neuroimage*, 23(4):1494–1506, 2004. 53

[45] C. Grefkes, P. H. Weiss, K. Zilles, and G. R. Fink. Crossmodal processing of object features in human anterior intraparietal cortex: An fMRI study implies equivalencies between humans and monkeys. *Neuron*, 35(1):173–184, 2002. 54

[46] F. Guerin. Learning like a baby: a survey of artificial intelligence approaches. *The Knowledge Engineering Review*, 26(02):209–236, 2011. 36

[47] F. Guerin and D. McKenzie. A Piagetian model of early sensorimotor development. In *Proceedings of the Eighth International Conference on Epigenetic Robotics*, Brighton, UK, 2008. 48

[48] F. Guerin and A. Starkey. Applying the schema mechanism in continuous domains. In *Proceedings of the Ninth International Conference on Epigenetic Robotics*, Venice, Italy, 2009. 48

[49] V. V. Hafner and G. Schillaci. From field of view to field of reach-could pointing emerge from the development of grasping? In *Proceedings of the IEEE Joint International Conference on Development and Learning, and Epigenetic Robotics*, Frankfurt, Germany, 2011. IEEE. 85

[50] S. Harnad. The symbol grounding problem. *Physica D*, 42(1-3):335–346, 1990. 43

[51] 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. 47

[52] M. Hülse, S. McBride, J. Law, and M. H. Lee. Integration of active vision and reaching from a developmental robotics perspective. *Autonomous Mental Development, IEEE Transactions on*, 2(4):355–367, 2010. 93

[53] J. M. Iverson and M. K. Fagan. Infant vocal–motor coordination: Precursor to the gesture–speech system? *Child Development*, 75(4):1053–1066, 2004. 39

[54] J. M. Iverson and S. Goldin-Meadow. Gesture paves the way for language development. *Psychological Science*, 16(5):367–371, 2005. 41

[55] J. M. Iverson, A. J. Hall, L. Nickel, and R. H. Wozniak. The relationship between reduplicated babble onset and laterality biases in infant rhythmic arm movements. *Brain and language*, 101(3):198–207, 2007. 39

[56] I. Kant. *Critique of pure reason, second edition.* Reprinted by Cambridge University Press, 1787 / 1999. 45

[57] E. Kohler, C. Keysers, M. Umiltà, L. Fogassi, V. Gallese, and G. Rizzolatti. Hearing sounds, understanding actions: action representation in mirror neurons. *Science*, 297(5582):846–848, 2002. 54

[58] N. Krüger, C. Geib, J. Piater, R. Petrick, M. Steedman, F. Wörgötter, A. Ude, T. Asfour, D. Kraft, D. Omrcen, A. Agosti, and R. Dillmann. Object-action complexes: Grounded abstractions of sensory-motor processes. *Robotics and Autonomous Systems*, 59(10):740 – 757, 2011. 48

[59] Y. Kuniyoshi and S. Sangawa. Early motor development from partially ordered neural-body dynamics: experiments with a cortico-spinal-musculo-skeletal model. *Biological cybernetics*, 95(6):589–605, 2006. 35

[60] J. Law, M. H. Lee, M. Hülse, and A. Tomassetti. The infant development timeline and its application to robot shaping. *Adaptive Behavior*, 19(5):335–358, 2011. 35

[61] M. H. Lee. Intrinsic activitity: from motor babbling to play. In *Proceedings of the IEEE Joint International Conference on Development and Learning, and Epigenetic Robotics*, Frankfurt, Germany, 2011. IEEE. 37

[62] M. H. Lee, J. Law, P. Shaw, and M. T. Sheldon. An infant inspired model of reaching for a humanoid robot. In *IEEE Joint International Conference on Development and Learning, and Epigenetic Robotics*, San Diego, USA, 2012. IEEE. 93

[63] M. H. Lee, Q. Meng, and F. Chao. Developmental learning for autonomous robots. *Robotics and Autonomous Systems*, 55(9):750–759, 2007. 38

[64] M. H. Lee, Q. Meng, and F. Chao. Staged competence learning in developmental robotics. *Adaptive Behavior*, 15(3):241, 2007. 38

[65] J. Lindblom and T. Ziemke. Social situatedness of natural and artificial intelligence: Vygotsky and beyond. *Adaptive Behavior*, 11(2):79, 2003. 40

[66] J. L. Locke, K. E. Bekken, L. Mcminnlarson, and D. Wein. Emergent control of manual and vocal-motor activity in relation to the development of speech. *Brain and Language*, 51(3):498–508, 1995. 39

[67] M. Lungarella, G. Metta, R. Pfeifer, and G. Sandini. Developmental robotics: a survey. *Connection Science*, 15(4):151–190, 2003. 36

[68] A. W. MacDonald, J. D. Cohen, V. A. Stenger, and C. S. Carter. Dissociating the role of the dorsolateral prefrontal and anterior cingulate cortex in cognitive control. *Science*, 288(5472):1835–1838, 2000. 53

[69] B. F. M. Marino, P. M Gough, V. Gallese, L. Riggio, and G. Buccino. How the motor system handles nouns: a behavioral study. *Psychological Research*, pages 1–10, 2011. 55

[70] N. Masataka. From index-finger extension to index-finger pointing: Ontogenesis of pointing in preverbal infants. In Sotaro Kita, editor, *Pointing: Where language, culture, and cognition meet*, pages 69–84. Lawrence Erlbaum Associates, 2003. 41

[71] C. B. Mervis and J. Bertrand. Acquisition of the novel name–nameless category (n3c) principle. *Child development*, 65(6):1646–1662, 1994. 43

[72] G. Metta, G. Sandini, D. Vernon, L. Natale, and F. Nori. The iCub humanoid robot: an open platform for research in embodied cognition. In *Proceedings of the 8th Workshop on Performance Metrics for Intelligent Systems*, pages 50–56. ACM, 2008. 89

[73] F. A. Middleton and P. L. Strick. Basal ganglia and cerebellar loops: motor and cognitive circuits. *Brain Research Reviews*, 31(2):236–250, 2000. 51

[74] K. Mourao, L. Zettlemoyer, R. P. A. Petrick, and M. Steedman. Learning STRIPS operators from noisy and incomplete observations. In *Proceedings of the 28th Conference on Uncertainty in Artificial Intelligence*, Catalina Island, USA, 2012. 48

[75] R. Mukamel, A. D. Ekstrom, J. Kaplan, M. Iacoboni, and I. Fried. Single-neuron responses in humans during execution and observation of actions. *Current Biology*, 20(8):750–756, 2010. 54

[76] A. Murata, L. Fadiga, L. Fogassi, V. Gallese, V. Raos, and G. Rizzolatti. Object representation in the ventral premotor cortex (area F5) of the monkey. *Journal of Neurophysiology*, 78(4):2226–2230, 1997. 55

[77] A. Murata, V. Gallese, G. Luppino, M. Kaseda, and H. Sakata. Selectivity for the shape, size, and orientation of objects for grasping in neurons of monkey parietal area aip. *Journal of Neurophysiology*, 83(5):2580–2601, 2000. 54

[78] D. K. Oller, L. A. Wieman, W. J. Doyle, and C. Ross. Infant babbling and speech. *Journal of Child Language*, 3(1):1–11, 1976. 39

[79] P. Y. Oudeyer and F. Delaunay. Developmental exploration in the cultural evolution of lexical conventions. In *Proceedings of the Eighth International Conference on Epigenetic Robotics*, Brighton, UK, 2008. 42

[80] P. Y. Oudeyer and F. Kaplan. Discovering communication. *Connection Science*, 18(2):189–206, 2006. 37

[81] P. Y. Oudeyer and F. Kaplan. What is intrinsic motivation? a typology of computational approaches. *Frontiers in Neurorobotics*, 1, 2007. 37

[82] P. Y. Oudeyer and F. Kaplan. How can we define intrinsic motivation? In *Proceedings of the Eighth International Conference on Epigenetic Robotics*, Brighton, UK, 2008. 37

[83] P. Y. Oudeyer, F. Kaplan, and V. V. Hafner. Intrinsic motivation systems for autonomous mental development. *Evolutionary Computation, IEEE Transactions on*, 11(2):265–286, 2007. 37

[84] E. Oztop and M. Arbib. Schema design and implementation of the grasp-related mirror neuron system. *Biological Cybernetics*, 87(2):116–140, 2002. 55

[85] E. Oztop, M. Kawato, and M. Arbib. Mirror neurons and imitation: A computationally guided review. *Neural Networks*, 19(3):254–271, 2006. 55

[86] M. Pardowitz and R. Dillmann. Towards life-long learning in household robots: The piagetian approach. In *Proceedings of the 6th IEEE International Conference on Development and Learning*, pages 88–93, London, UK, 2007. 35

[87] F. S. Perotto and L. O. Álvares. Learning regularities with a constructivist agent. In *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, pages 807–809. ACM, 2006. 48

[88] F. S. Perotto and L. O. Álvares. Incremental inductive learning in a constructivist agent. *Research and Development in Intelligent Systems XXIII*, pages 129–142, 2007. 48

[89] F. S. Perotto, J. Buisson, and L. O. Álvares. Constructivist anticipatory learning mechanism (CALM) - dealing with partially deterministic and partially observable environments. In *Proceedings of the Seventh International Conference on Epigenetic Robotics*, San Diego, USA, 2007. 48

[90] J. Piaget. *The construction of reality in the child*, volume 20. Reprinted by Routledge, 1937 / 1999. 33

[91] J. Piaget, J. Tomlinson, and A. Tomlinson. *The child's conception of the world*. Routledge & Kegan Paul, 1929. 33

[92] D. J. Povinelli, J. E. Reaux, D. T. Bierschwale, A. D. Allain, and B. B. Simon. Exploitation of pointing as a referential gesture in young children, but not adolescent chimpanzees. *Cognitive Development*, 12(4):423–461, 1997. 41

[93] P. Redgrave, T. J. Prescott, and K. Gurney. The basal ganglia: a vertebrate solution to the selection problem? *Neuroscience*, 89(4):1009–1024, 1999. 51

[94] M. B. Ring. CHILD: A first step towards continual learning. *Machine Learning*, 28(1):77–104, 1997. 47

[95] G. Rizzolatti, L. Fadiga, V. Gallese, and L. Fogassi. Premotor cortex and the recognition of motor actions. *Cognitive brain research*, 3(2):131–141, 1996. 54

[96] R. M. Ryan and E. L. Deci. Intrinsic and extrinsic motivations: Classic definitions and new directions. *Contemporary educational psychology*, 25(1):54–67, 2000. 37

[97] E. Şahin, M. Çakmak, M. R. Doğar, E. Uğur, and G. Üçoluk. To afford or not to afford: A new formalization of affordances toward affordance-based robot control. *Adaptive Behavior*, 15(4):447–472, 2007. 49

[98] G. Sandini, G. Metta, and D. Vernon. The iCub cognitive humanoid robot: An open-system research platform for enactive cognition. *50 years of artificial intelligence*, pages 358–369, 2007. 89

[99] J. Schmidhuber. Developmental robotics, optimal artificial curiosity, creativity, music, and the fine arts. *Connection Science*, 18(2):173–187, 2006. 37

[100] J. R. Searle. Minds, brains, and programs. *Behavioral and Brain Sciences*, 3(3):417–424, 1980. 43

[101] M. T. Sheldon and M. H. 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, Ann Arbor, USA, 2010. IEEE. 107

[102] M. T. Sheldon and M. H. Lee. PSchema: A developmental schema learning framework for embodied agents. In *Proceedings of the IEEE Joint International Conference on Development and Learning, and Epigenetic Robotics*, Frankfurt, Germany, 2011. IEEE. 47, 59

[103] S. Singh, A. G. Barto, and N. Chentanez. Intrinsically motivated reinforcement learning. In *Proceedings of the 17th Annual Conference on Advances in Neural Information Processing Systems*, 2005. 37

[104] L. B. Smith and C. Yu. Infants rapidly learn words from noisy data via cross-situational statistics. In *Proceedings of the 29th Annual Conference of the Cognitive Science Society*, volume 27, 2007. 43

[105] L. Steels. Evolving grounded communication for robots. *Trends in Cognitive Sciences*, 7(7):308–312, 2003. 42

[106] L. Steels, F. Kaplan, A. McIntyre, and J. Van Looveren. Crucial factors in the origins of word-meaning. *The Transition to Language*, 2(1):4–2, 2002. 42

[107] M. Tettamanti, G. Buccino, M. C. Saccuman, V. Gallese, M. Danna, P. Scifo, F. Fazio, G. Rizzolatti, S. F. Cappa, and D. Perani. Listening to Action-related Sentences Activates Fronto-parietal Motor Circuits. *Journal of Cognitive Neuroscience*, 17(2):273–281, 2005. 54

[108] V. Tikhanoff, A. Cangelosi, and G. Metta. Language understanding in humanoid robots: icub simulation experiments. *IEEE Transactions on Autonomous Mental Development*, 3(1):17–29, 2011. 42

[109] M. Tomasello, M. Carpenter, and U. Liszkowski. A new look at infant pointing. *Child Development*, 78(3):705–722, 2007. 41

[110] M. Umiltà, E. Kohler, V. Gallese, L. Fogassi, L. Fadiga, C. Keysers, and G. Rizzolatti. I know what you are doing a neurophysiological study. *Neuron*, 31(1):155–165, 2001. 55

[111] L. S. Vygotsky. Mind in society: Development of higher psychological processes. Harvard University Press, 1978. 35, 40, 107

[112] D. Wood, J. S. Bruner, and G. Ross. The role of tutoring in problem solving. *Journal of child psychology and psychiatry*, 17(2):89–100, 1976. 35

[113] C. Yu and L. B. Smith. Rapid word learning under uncertainty via cross-situational statistics. *Psychological Science*, 18(5):414–420, 2007. 43

[114] C. Yu, L. B. Smith, K. Klein, and R. M. Shiffrin. Hypothesis testing and associative learning in cross-situational word learning: Are they one and the same. In *Proceedings of the 29th Annual Conference of the Cognitive Science Society*, pages 737–742, 2007. 43

[115] J. Zlatev. The epigenesis of meaning in human beings, and possibly in robots. *Minds and Machines*, 11(2):155–195, 2001. 43

Appendix A

Papers

A.1 IEEE International Conference on Development and Learning 2010[1]

[1]In the third paragraph of the communication section the terms *'proto-imperative'* and *'proto-declarative'* have been mistakenly reversed. As such that section should read: *'The developmental progression previously outlined for robotic systems currently emerges proto-imperative pointing before going on to make the leap towards treating this in a proto-declarative manner in some of the later stages, while this may not be the exact progression experienced by children it provides a simpler mechanism for a robot to learn pointing gestures, albeit in a less rich developmental context.'*

A developmental approach to the emergence of communication in socially situated 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—This paper reports on a developmental approach to the learning of communication in embodied agents, taking inspiration from child development and recent advances in the understanding of the mirror neuron system within the brain. We describe a part of the ROSSI project which focuses upon gestural communication in the form of pointing. We are examining the idea that pointing may be a key step towards simple spoken communication and exploring the internal representations that may be formed during this process.

The possible developmental stages leading to proto-imperative pointing actions in a robotic system are outlined, and how this may be built upon to result in an understanding of two word speech is discussed. The learning mechanism is based around Piagetian schema learning whilst the developmental path follows a mixture of Piagetian and Vygotskian theories.

Index Terms—Language Development, Human-Robot Interaction, Embodied Cognition, Grounding of Knowledge and Representations

I. INTRODUCTION

The developmental approach to robotics in which systems attempt to mimic similar stages of development to a human infant has so far had little application to the possible emergence of communication and symbol grounding in robotic systems.

In this paper we explore the emergence of early gestural communication as a side-effect of sensorimotor robot learning and how this may be used to boot-strap simple linguistic communication for robotic systems. While we take inspiration from infant development we do not claim to be accurately modelling human development.

The success of this approach depends upon the willing co-operation of other social agents to aid the robot in its learning. The robot is not imbued with any innate theory of communication, so if it never experiences communicative acts from other agents it will be unable to learn to communicate itself.

This approach allows symbolic meaning in the form of language to be strongly rooted in the sensorimotor experience of the agent, with the various concepts involved in communication arising out of interaction with the environment and other agents. The same learning framework is used throughout all stages of development (although not all aspects of the framework are used at all stages) allowing more advanced concepts to be grounded in simpler ones from earlier developmental stages.

A. Developmental stages

The following seven stages outline a possible robotic developmental progression leading from a "new born" state to simple linguistic communication. This paper focuses mostly on the stages leading to pointing gestures, with future work extending this to build up to speech. This progression, especially in the latter stages, is based heavily upon that described by Iverson and Goldin-Meadow [10], discussed further in section II-A (communication).

1) Motor babbling: In this initial stage the robot has had no prior experience of the world or of its own body. It performs spontaneous motor actions in order to discover the properties of its motor systems and its anatomical constraints.

2) Motor vision mapping: The movements learnt in the previous stage are then mapped to the changes they create in the robot's vision system, this allows it to move its arm to touch (or point towards) an object detected visually. While the focus in this paper is on visual mappings this could equally be applied to other sensor modalities.

3) Failed grasping leading to pointing: In attempting to touch objects that lie outside of the work-envelope of the robot it will incidentally perform what looks, to a human observer, like a pointing motion. Through assistance from a human observer, fetching the indicated object for the robot, the robot's representation of this action moves away from being a direct attempt at manipulating the world towards an attempt at social communication.

4) Recognising pointing in others: Using a goal directed approach based on mirror neuron theory, the robot is then able to learn to reciprocate, providing objects to humans (or other robots) when they are indicated. This allows the structures necessary for a simple give/take conversation to emerge prior to the introduction of language.

5) Complementary one word speech with pointing: The robot is then given auditory input (reduced to a text token by speech recognition software) whilst it points at objects, or whilst it sees a human or other robot point at an object. This input is directly related to the object being indicated, for example the word "ball" or "block".

6) Supplementary one word speech with pointing: In this stage the auditory input relates to the action being indicated rather than the object itself. The pointing action has been used

in the preceding stages as a request for the indicated object, so the word "give" becomes associated with this action.

7) Two word sentences: Finally the learnt speech can be combined to allow the robot to form and understand two word sentences of the form "give block", replacing the pointing behaviour from the earlier stages.

II. BACKGROUND AND RELATED WORK

A. Communication

Vygotsky suggested that pointing develops out of a failed grasping behaviour in which the child attempts to reach for an object which is too far away, the parent interprets this as the child pointing at a desired object and as such fetches the object for the child, thus associating a new meaning with the act of reaching for a distant object [27], [14]. Initially all social meaning in this act is inferred entirely by the parent, the infant is making a real attempt to reach the object and failing, but through the actions of the parent the infant comes to associate the same communicative meaning.

This has been classified by many researchers as *proto-imperative pointing* or *ritualised grasping*, used by the child to indicate an object of desire to a nearby adult, and typically emerges at around 10-12 months. On average 3 months [5] after the emergence of proto-imperative pointing the child has also learnt to perform *proto-declarative pointing* which is used to acquire joint attention on an object with an adult.

There is however evidence presented by Masataka [17] to indicate that proto-imperative pointing and proto-declarative pointing may follow different developmental paths, with proto-imperative pointing actually arising out of index finger extension for the purposes of object exploration. The developmental progress previously outlined for robotic systems currently emerges proto-declarative pointing before going on to make the leap towards treating this in a proto-imperative manner in some of the later stages, while this may not be the exact progression experienced by children it provides a simpler mechanism for a robot to learn pointing gestures, albeit in a less rich developmental context. In addition to this Tomasello, et al. [25] show that infants may possess a much deeper social understanding at this stage than previously thought, able to communicate a great deal through pre-linguistic gestures such as pointing.

Butterworth [4] provides various evidence supporting the theory that gesture is nearly universal on the road to further language development.

Iverson and Goldin-Meadow [10] describe the early developmental path of infants learning to communicate verbally. They show that in most cases infants follow a consistent progression from pointing to two word speech, as described in the later stages of the previously outlined developmental progression.

B. Neuroscience

The mirror neuron system was first discovered in the brains of monkeys [7], [22] and later studies showed a similar system at work in the human brain. A mirror neuron is a neuron which fires both upon the execution of an action and upon the observation of another agent performing the action. Each mirror neuron is paired with a canonical neuron, however the canonical neuron is only activated during the execution of an action and not during its observation. This has prompted speculation that the mirror neuron system may have been crucial in the evolution of language [1].

Tettamanti, et al. [24] show that listening to action related sentences can trigger a mirror neuron response in humans and Kohler, et al. [11] have previously found that a noise related to an action can trigger a response in monkeys. This adds further weight to the idea that the mirror neuron system encodes action content at an abstract level and that this content can be activated auditorily. This suggests that language is strongly linked to the sensorimotor system.

A study by Buccino, et al. [3] suggests that mirror neuron responses only occur for actions that the observer can duplicate. For example humans watching a dog biting will show front-parietal activity, while they will not when watching a dog bark. This also shows that the mirror neuron system generalises to different species, possibly suggesting that the goal of the action has a much greater effect than the observation of the action itself.

The goal directed nature of mirror neurons is further reinforced by a study by Umiltà, et al. [26] in which the neural response from monkeys was measured when they observed the experimenter grasping an object and when they observed a mimed grasp with no object present. It was found that the mimed grasp produced no response, while the real grasp did. It was also found that if the view of the object was occluded so the final stage of the grasp wasn't visible then some response was still produced, suggesting that the goal was being inferred from the action.

Oztop and Arbib [19] hypothesise that the mirror neuron system may have evolved to provide feedback for visually directed grasping with the social usage being an exaptation[1] occurring when this became applied to the hands of others.

Oztop, Kawato and Arbib [20] provide a computationally guided review of mirror neuron literature and provide box diagrams of a computational model called the MNS model. Bonaiuto, et al. [2] have made attempts to extend this model, creating a more comprehensive version titled MNS2. Small sections of this model have been implemented and tested, but the model as a whole remains largely theoretical.

C. Robotics and artificial intelligence

Drescher [6] suggests 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 sensor values specified in that schema's prediction.

---

[1]An exaptation being the exploitation of an evolutionary adaptation to serve a different purpose than the one it initially developed for.

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.

Holmes and Isbell [9] 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.

Guerin [8] has since used this approach in a simple simulated robotic environment, but as yet little work has been performed using this technique on a physical robot.

Perotto, et al. [21] 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 to the environment providing a more direct mapping between the environment and the agent's perceptions than a state based environment.

Lee, et al. [13], [12] discuss the use of a Lift Constraint, Act, Saturate (LCAS) 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. This approach is similar to the scaffolding [15] performed by parents when helping children to learn in that the staged constraints placed upon the system's sensory input provides a framework that guides the robot through its development. Once there is little novel input being found at one stage of learning a constraint is 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.

Marjanovic, et al. [16] introduce a motor-vision mapping system that learns to perform pointing motions towards visual targets. Our system differs from this in that the one presented by Marjanovic has an explicit goal of pointing, while in our system this behaviour emerges as a side effect of other developmental processes occurring at the same time and as a product of social interaction.

Steels, et al. [23] show that the concept formation process of agents must be based on similar sensor input and result in similar conceptual repertoires for communication to develop in a population of agents. It also shows that once a lexical system is in place it can overcome the randomness inherent in verbal communication.

Oudeyer and Kaplan [18] explore the intrinsic motivation of language learning rooted in play and curiosity, using a framework based around Vygotsky's zone of proximal development [28] (although this is termed *"progress niches"* within this system). It shows how an intrinsic motivation system can allow a robot to self-organise its learning process.

III. HARDWARE CONFIGURATION

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.

This hardware setup can be seen in figure 1.

Fig. 1. The current hardware configuration.

Use of this learning framework in the context of a more complex system, involving many more degrees of freedom, is discussed briefly in section VII (future work).

IV. THE SOFTWARE FRAMEWORK

The system consists of two main components, the schema memory and the developmental controller. The developmental controller determines the goal of the system based on the current excitation level and motivation, as well as handling the reduction of complexity in sensory input based upon the current learning stage. These sensor values are then passed to the schema memory along with either an action or a desired goal state, the resulting schema(s) are then executed and the results stored for use in judging their suitability for future tasks.

A. Schema learning framework

A similar approach to Drescher's schema learning is used to achieve the desired learning behaviour, albeit with a number of modifications from Drescher's original design to make the technique more applicable to robotics. Unlike Drescher's binary system or Holmes and Isbell's continuous value system the schema framework makes use of discrete sensor values

made up of sensorimotor fields which reduce the complexity of the sensor input and motor output.

While a very symbolic schema representation has been chosen here a neural implementation should give similar results, however we believe that a more explicit symbolic representation lends itself to easier analysis of the resulting generated internal structures.

B. Schema chaining

The linking of pre-conditions and post-conditions from different schemas (*"schema chaining"*) creates a traversable network representing different world states and the actions required to move between these states, as illustrated in figure 2. Without schema chaining the robot's interest in unreachable objects would decrease as it failed to reach them. Schema chaining allows for cases in which the feedback of an action isn't instantaneous to still be recognised as being useful. Thus making the entire series of actions required to point at an object, wait for another agent to move the object then touch the object interesting to the robot, despite the reward (touching the object) being at the end of the chain of actions.
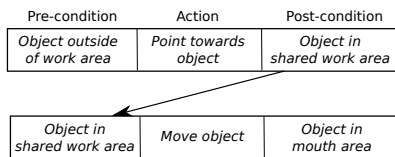

Fig. 2. A high level example of schema chaining, allowing the robot to gain access to an object that would otherwise be outside of its reach through communication with another agent.

C. Tracking of Schema Probabilities

The schema framework keeps track of the observed probabilities of the post-conditions of each schema, allowing it to predict the most likely outcome. The storing of probabilities for the likelihood of individual items, instead of the probability of the schema as a whole being successful (as proposed by Drescher) allows the system to select the best action for achieving its target goal, regardless of the likelihood of less interesting side-effects of the action. For example it makes little sense for the system to care how likely it is that a particular block is moved when the goal of the action is just to move the arm to a specific location, the movement of a block (or lack thereof) is merely an uninteresting side-effect in the context of this particular goal.

D. Schema Generalisation

The system periodically attempts to generalise its existing schemas, the specific schemas from which these generalisations arise are retained and when an attempted action does not meet the expected outcome from a general schema a new specific schema is created, allowing future attempts at refining the generalisation with the added information from the failed tests. When performing an action a specific schema is preferred over a generalised schema if one exists that matches.

For example when seeing a number of specific schemas along the lines of *[object in field 5] / [move arm to field 5] / [object in field 5, finger in field 5, touching]* the system will generate a general schema of the form *[object in field $x] / [move arm to field $x] / [object in field $x, finger in field $x, touching].*

E. Mirror Neuron Influenced Schema Learning

To enable the schema system to mimic the behaviour of the human/primate mirror neuron system it is split up in to two distinct classes, traditional schemas, with pre-condition, action and post-condition components, and *"perceptual schemas"* which lack an action component and are used for observing the actions of another agent. These classes are linked together when a perceptual schema and a traditional schema have matching post-conditions allowing the observation of other agents performing an action to be associated with the observer's motor schema for that same action.

Experiments with monkeys have suggested that their mirror neuron system is largely dependent on the goal of an action [7], [26], in that mirror neurons will fire when the monkey observes the experimenter grasping an object, but will not fire, or only fire very weakly, when they observe a *"pantomime gesture"*, in which the experimenter performs the same action but without an object present. The system mimics this behaviour by using post-conditions as the linking mechanism between the different schema classes, for example the visual input of watching another agent move an object can be associated with the motor behaviour performed when the observer is themselves moving an object.

In addition to providing a mechanism for recognising the actions of other agents, this also provides part of the framework necessary for spoken language. This will allow the linking of auditory observations to actions and other observations, helping the system move from a "motor-meaning" based representation to a "symbolic-meaning" which is one of the key differences between Piaget's stage 3 and stage 4 infant.

F. Developmental controller

A control program has been developed that makes use of the schema framework. The control system has two different modes of operation, a "play" mode, in which it randomly executes schemas based on their predicted excitement and reward levels and a "task" mode, in which it can perform more goal directed actions.

The resolution of the robot's inputs are reduced by the control program to speed learning. The possible joint configurations are reduced to typically 200-300 combinations depending on the robot configuration (referred to as the *"motor fields"*). This is achieved by limiting the robot to the use of two joints, each moving in 10 degree increments, a third joint becomes accessible when the robot is at the outermost limit of its two joint work envelope allowing the end-effector to be moved outwards tracing a vertical arc to allow for pointing. The visual system is similarly divided in to circular fields each with a 10 pixel radius, referred to as "visual fields", a specific type of sensorimotor field. A new visual field is created each

time the robot observes an object outside of its current fields, with the centre point of this object forming the centre of the field. The fields are initially discovered by the robot exercising its previously learned motor schemas and observing its end-effector entering the different visual fields [12].

The controller implements a Lift-Constraint, Act, Saturate (*"LCAS"*) [13] based approach to staged learning. Additional constraints are added to the robot's sensory input, these constraints are lifted as the robot becomes habituated to its current level of development. The point at which these constraints should be lifted is determined by the system's excitation level. This excitation level is also used to decide which actions to perform next, for example an action which would cause a new schema to be created would be considered more exciting than the execution of an existing schema. Whether an action is executed or not depends on whether or not it is above a certain threshold below the global excitation level. This means that if most of the actions the robot is performing are creating new schemas then it is unlikely to execute any existing schemas, however once there are fewer new schemas to discover it will begin to re-activate existing schemas with a preference for those which have had the fewest activations. Figure 3 shows the system reaching a plateau during the motor babbling stage, once this has been reached the constraint on the vision system is lifted and the robot begins to map visual input to its existing schemas, the number of schemas does not begin to rise again until the robot's environment is made more complicated through the introduction of wooden blocks for it to manipulate. In addition to novelty-triggered excitation the robot also receives a reward for successfully touching an object (making such actions more exciting). This biases it towards actions that may result in contact when an object is present, this helps to speed learning by focusing the robot's attention on actions more likely to lead towards the desired behaviour.
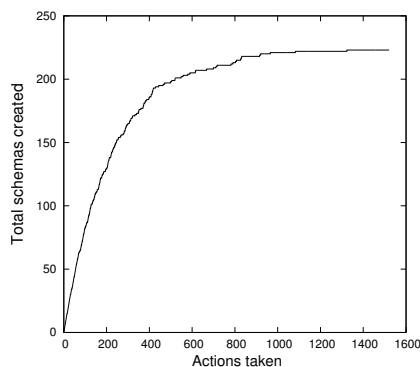
Fig. 3. As time progresses the robot discovers less novel situations requiring new schemas and instead exercises existing schemas to test their reliability. Once the developmental controller detects the saturation of this stage of development the complexity of the input to the system is increased and the schemas can be further refined.

V. REPRESENTATIVE SCHEMAS

A. Motor babbling

Initially very basic schemas are created with no context, representing only actions.

Pre-conditions	Action	Post-conditions
	Motor action joint1=0.69 joint2=0.87 joint3=0	

B. Motor vision mapping

Later the most basic visual result of these actions (the end effector appearing in a different field) are added as post-conditions.

Pre-conditions	Action	Post-conditions
	Motor action joint1=0.69 joint2=0.87 joint3=0	End effector in field 7

C. Touching objects

Next the robot is given a few examples of touching objects in different positions.

Pre-conditions	Action	Post-conditions
Object in field 3	*Motor action* joint1=0.23 joint2=0.43 joint3=0	Object in field 3 End effector in field 3 Touching

Once a number of examples along these lines have been viewed this gets generalised, to give a schema which represents touching an object in any position on the work surface.

Pre-conditions	Action	Post-conditions
Object in field $x	*Target action* End effector in field $x	Object in field $x End effector in field $x Touching

In these generalised schemas we see the use of *"target actions"* replacing direct motor actions, rather than causing a direct change in the robot's configuration they represent a target set of post-conditions that should be achieved (which is a subset of the post-conditions of the main schema). This allows the generalisation to occur across the pre-conditions, post-conditions and action with consistent variables.

D. Pointing counter-examples

In the case of pointing the system attempts to execute the above generalised touching schema but fails, generating a specific counter-example. Specific schemas are always preferred over generalised schemas if both fulfil the same conditions. This allows the system to learn where its generalisation fails and create schemas that work in those

situations.

Pre-conditions	Action	Post-conditions
Object in field 203	*Target action*	Object in field 56
	End effector in field 203	End effector in field 203

Due to the random, non-contiguous nature of the visual fields the current system must learn each pointing location individually. In future experiments the system will be tested with a predefined contiguous visual space allowing the generation of a second generalised schema representing the pointing space with a conditional observation such as *"$x > 134"*, where field 134 marks the shift from touching to pointing.

VI. POINTING MECHANISM

The controller takes the system through two learning stages to create a mapping between the motor system and the vision system. This mapping allows the robot to move its end effector in to a desired visual field, which can then be used for allowing it to interact with objects (both by physically touching them and moving them around itself and by pointing at them for communication).

The first stage of this process is akin to Piaget's first stage infant, the robot goes through a period of *"motor babbling"*, where it exercises all possible joint configurations and creates schemas representing these actions. It receives no feedback from these actions, merely generating a base set of schemas that abstract higher level schemas away from explicit joint commands, allowing them to instead refer to existing schemas as their action components.

In the second stage the vision system is made available to the robot and it begins to associate visual context with the existing motor schemas. This is similar to hand fixation in an infant. This stage is visualised in figure 4. The robot executes the purely motor based schemas it has learnt in the previous stage and forms a new visual field whenever it sees its end effector outside of any existing fields, it then adds this as a new post-condition to the executed schema. The end effector is detected via the vision system, potentially it will add any changes in visible objects as post-conditions, however at this stage of the robots learning no other objects are presented to it.

Fig. 4. A visualisation of the visual fields, part-way through their discovery.

The systems operates primarily on the X-Y plane using 2 degrees of freedom, illustrated in figure 5(a). To enable the robot to point at objects outside of its work envelope it is able to slightly lift its end effector when at the furthest extent of its normal range of motion, shown in figure 5(b). Both of these planes are accessible to the robot throughout all stages of learning, so it first learns to position its end effector in the 'pointing' plane prior to any objects being introduced for it to point at as part of its random motor babbling and vision mapping stages.

It is important to note that this is not giving the robot a full 3D representation of the space it occupies as the robot still effectively lacks accurate depth perception, however for the purposes of this experiment that is unimportant and may indeed be congruous with a child's perception at this stage. If similar operations were performed on a system with more degrees of freedom the same outcome should be possible, with the added benefit that the system would be able to point to objects within its work envelope without touching them. We only constrain the system to 2 DoF to greatly simplify the lower level motor learning tasks.
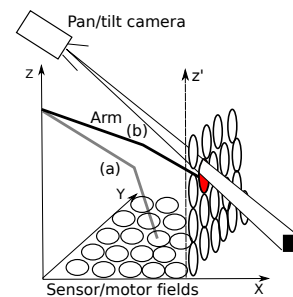
Fig. 5. *(a)* Direct manipulation of objects is only possible in the X-Y plane. *(b)* The robot's motion is extended in to the Z' plane allowing it to pointing to distant objects. This is a simplified example, rather than having two distinct vertical and horizontal planes the system operates on a manifold that curves up at the extremities tracing the outside of the robot's work envelope.

When first learning to point the robot views an object and moves its end effector to occupy the same visual field, using a generalised form of the schemas it has developed to allow it to touch objects (and so receive a reward). However in this case the schema does not successfully result in contact with the end effector, instead it results (from the perspective of a human observer) in a pointing motion towards the object. The robot is receiving no reward when failing to touch the object, however in the event that a human observer assists the robot by moving the object closer it leads to a chain of events which finally results in the robot touching the object and so being rewarded.

A. Morphological implications

This approach raises certain morphological implications. For a pointing gesture that a human would recognise to emerge from this technique the robot in question must itself have a roughly humanoid anatomy. Specifically it requires the robot's vision system to be positioned above the arm system looking out in the direction of action. Additionally for the pointing to

appear accurate the vertical distance between the vision system and the arm should not be too great.

All current testing has been performed with humans with prior knowledge that what they are about to view is intended as a pointing gesture, it might be interesting to investigate the effects this gesture has on people who do not already know what to look for. The anthropomorphic characteristics of the robot in question might play as large a part in this as the quality of the gesture itself. However for now this is outside the scope of the current investigation.

VII. FURTHER WORK

This paper deals primarily with the initial emergence of pointing behaviour and the stages preceding it. We are continuing with the later stages in the developmental progression, including the recognition of pointing from other agents and the transition to linguistic communication. Work on these aspects is ongoing.

We have implemented a neurally-inspired reaching/grasping model for a 7 DoF tactile sensing robot hand (Schunk GmbH & Co.) as part of the ROSSI project. The schema system is in the process of being integrated with this so that a wider range of possible actions and gestures may be investigated. In this configuration rather than dealing with the vision and motor system directly the schema system talks to an affordance based memory which processes object features and determines the appropriate joint configuration for grasping them, meaning the schema system can continue to operate at a fairly high, symbolic level while the affordance memory deals with the low level joint configuration in more detail. This system also has the capacity to recognise human hand positions via a data glove, which provides an ability to imitate humans and will allow us to determine more accurately when a human is pointing at an object. Schema learning adds a capacity for temporal reasoning and goal directed behaviour that is lacking in the current affordance based grasp system.

There is also the potential for further work focusing upon one robot having learnt this process with a human teacher and then going on to teach a second robot in a similar manner. This could be further extended to look at the implications on a larger population of robots and how social meanings might adapt due to slight changes in the teaching process from one robot to another, following a similar methodology to Steels, et al. [23].

In the current system the robot has no mechanism for perceiving the presence of another agent as there is assumed to always be a human present. If this facility were to be added in the future it would allow the robot to discover in which scenarios social acts are likely to be successful.

VIII. ACKNOWLEDGEMENTS

REFERENCES

[1] M. Arbib. The mirror system, imitation, and the evolution of language. *Imitation in Animals and Artifacts*, pages 229–280, 2002.

[2] J. Bonaiuto, E. Rosta, and M. Arbib. Extending the mirror neuron system model, I. *Biological Cybernetics*, 96(1):9–38, 2007.

[3] G. Buccino, F. Lui, N. Canessa, I. Patteri, G. Lagravinese, F. Benuzzi, C. A. Porro, and G. Rizzolatti. Neural circuits involved in the recognition of actions performed by nonconspecifics: An fMRI study. *Journal of Cognitive Neuroscience*, 16(1):114–126, 2004.

[4] G. Butterworth. Pointing is the royal road to language for babies. *Pointing: Where language, culture, and cognition meet*, pages 9–33, 2003.

[5] L. Camaioni, P. Perucchini, F. Muratori, and A. Milone. Brief report: A longitudinal examination of the communicative gestures deficit in young children with autism. *Journal of autism and developmental disorders*, 27(6):715–725, 1997.

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

[7] V. Gallese, L. Fadiga, L. Fogassi, and G. Rizzolatti. Action recognition in the premotor cortex. *Brain*, 119(2):593, 1996.

[8] F. Guerin and D. McKenzie. A Piagetian model of early sensorimotor development. *Proceedings from Epigenetic Robotics 2008*, 2008.

[9] 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.

[10] J. M. Iverson and S. Goldin-Meadow. Gesture Paves the Way for Language Development. *Psychological Science*, 16(5):367–371, 2005.

[11] E. Kohler, C. Keysers, M. Umiltà, L. Fogassi, V. Gallese, and G. Rizzolatti. Hearing sounds, understanding actions: action representation in mirror neurons. *Science*, 297(5582):846–848, 2002.

[12] M. H. Lee, Q. Meng, and F. Chao. Developmental learning for autonomous robots. *Robotics and Autonomous Systems*, 55(9):750–759, 2007.

[13] M. H. Lee, Q. Meng, and F. Chao. Staged competence learning in developmental robotics. *Adaptive Behavior*, 15(3):241, 2007.

[14] J. Lindblom and T. Ziemke. Social Situatedness of Natural and Artificial Intelligence: Vygotsky and Beyond. *Adaptive Behavior*, 11(2):79, 2003.

[15] M. Lungarella, G. Metta, R. Pfeifer, and G. Sandini. Developmental robotics: a survey. *Connection Science*, 15(4):151–190, 2003.

[16] M. Marjanovic, B. Scassellati, and M. Williamson. Self-taught visually guided pointing for a humanoid robot. In *From Animals to Animats 4: Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior*, pages 35–44, 1996.

[17] N. Masataka. From index-finger extension to index-finger pointing: Ontogenesis of pointing in preverbal infants. *Pointing: Where language, culture, and cognition meet*, pages 69–84, 2003.

[18] P. Y. Oudeyer and F. Kaplan. Discovering communication. *Connection Science*, 18(2):189–206, 2006.

[19] E. Oztop and M. Arbib. Schema design and implementation of the grasp-related mirror neuron system. *Biological Cybernetics*, 87(2):116–140, 2002.

[20] E. Oztop, M. Kawato, and M. Arbib. Mirror neurons and imitation: A computationally guided review. *Neural Networks*, 19(3):254–271, 2006.

[21] 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.

[22] G. Rizzolatti, L. Fadiga, V. Gallese, and L. Fogassi. Premotor cortex and the recognition of motor actions. *Cognitive brain research*, 3(2):131–141, 1996.

[23] L. Steels, F. Kaplan, A. McIntyre, and J. Van Looveren. Crucial factors in the origins of word-meaning. *The Transition to Language*, 2(1):4–2, 2002.

[24] M. Tettamanti, G. Buccino, M.C. Saccuman, V. Gallese, M. Danna, P. Scifo, F. Fazio, G. Rizzolatti, S. F. Cappa, and D. Perani. Listening to Action-related Sentences Activates Fronto-parietal Motor Circuits. *Journal of Cognitive Neuroscience*, 17(2):273–281, 2005.

[25] M. Tomasello, M. Carpenter, and U. Liszkowski. A new look at infant pointing. *Child Development*, 78(3):705–722, 2007.

[26] M. Umiltà, E. Kohler, V. Gallese, L. Fogassi, L. Fadiga, C. Keysers, and G. Rizzolatti. I Know What You Are Doing A Neurophysiological Study. *Neuron*, 31(1):155–165, 2001.

[27] L. S. Vygotsky. Mind In Society. page 56, 1978.

[28] L. S. Vygotsky. Mind In Society. pages 84–91, 1978.

A.2 IEEE International Conference on Development and Learning, and Epigenetic Robotics 2011

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:

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

$$(1)$$

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 ∪ 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 backplane. 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.
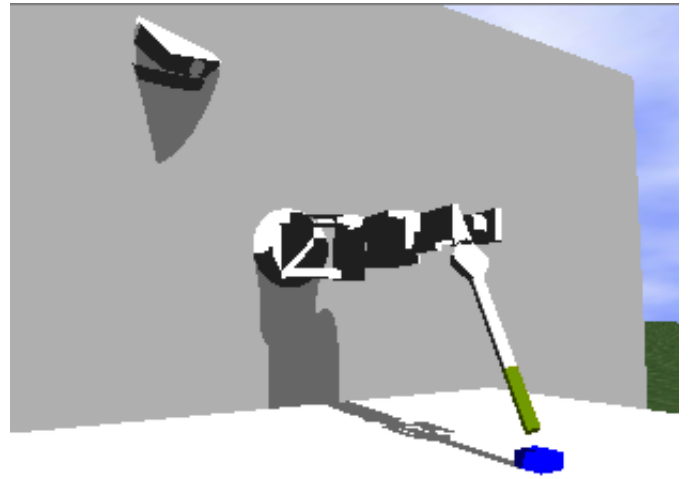

Fig. 5. The simulation environment showing the arm pointing at an object placed slightly outside of the robot's work envelope.

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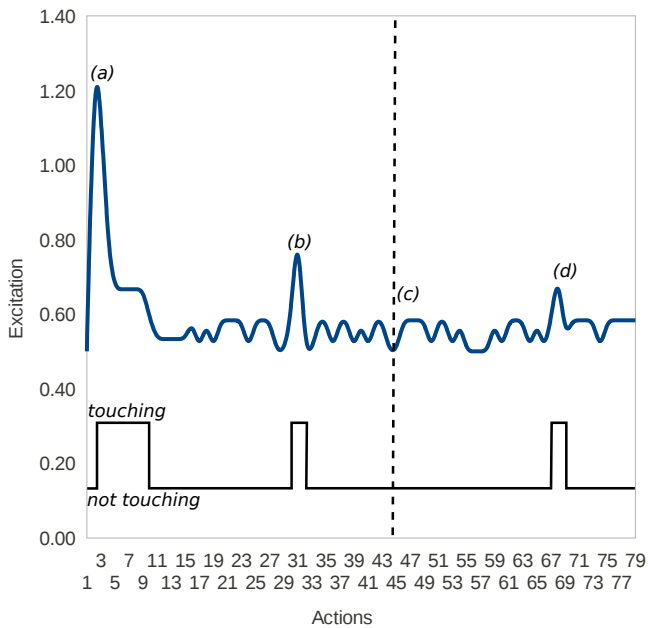
Fig. 6. *Top:* Level of excitement provided by the most excited schema at each time-step during the language experiments. *Bottom* The type of schema being executed by the system, with the high state representing a touching schema and the low state being any other schema.

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 relearnt 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

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.

A.3 IEEE International Conference on Development and Learning, and Epigenetic Robotics 2012

An Infant Inspired Model of Reaching for a Humanoid Robot

Mark Lee, James Law *Member, IEEE,* Patricia Shaw, and Michael Sheldon

Abstract—**Infants demonstrate remarkable talents in learning to control their sensor and motor systems. In particular the ability to reach to objects using visual feedback requires overcoming several issues related to coordination, spatial transformations, redundancy, and complex learning spaces, that are also challenges for robotics.**

The development sequence from tabula rasa to early successful reaching includes learning of saccade control, gaze control, torso control, and visually elicited reaching and grasping in 3D space. This sequence is an essential progression in the acquisition of manipulation behaviour.

In this paper we outline the biological and psychological processes behind this sequence, and describe how they can be interpreted to enable cumulative learning of reaching behaviours in robots. Our implementation on an iCub robot produces reaching and manipulation behaviours from scratch in around 2.5 hours. We show snapshots of the learning spaces during this process, and comment on how timing of stage transition impacts on learning.

I. INTRODUCTION

REACHING in humans requires the coordination of several different muscle groups controlling the shoulder, elbow, and wrist. Each of these requires relations to be made between the range of proprioceptively sensed positions and the muscle movements needed to reach those positions. Furthermore, reaching to seen objects requires the space of possible reach positions to be mapped onto the visual space perceived by the eye, but this is not straightforward as multiple arm poses may be available to reach each seen position [1].

These issues pose problems for reach-learning in humanoid robots. Multiple kinematically dependent joints create large learning spaces; visual- and joint-spaces are not topographically related, requiring some kind of transformation; redundancy creates multiple joint poses for reaching to point targets, and these cause difficulties in generating smooth reaching trajectories without discontinuities. There have been many studies and experiments on robot reaching, using both neural models and AI based methods, e.g. [2], but very few perform hand/eye coordination learning on complex kinematics in real time without prior training.

We present an approach to reach learning in humanoid robotics that draws heavily from the psychological literature and is inspired by the development and behaviour of very early infants. We identify several key factors that we consider important principles to be included in our models:

- *Motor babbling.* This is spontaneous, internally motivated, action that generates sensorimotor data during infancy. We show that it is not random activity but is functional in relating previous action to current and new

sensory-motor patterns. This has close links to the role of play behaviour.
- *Proprioception.* Proprioception develops in the pre-natal stages and, along with motor babbling, is likely to enable learning of muscle control. Proprioception develops before vision and visual guidance in reaching, and is a key factor in learning reaching motions.
- *Proximal to distal development.* Infant development follows a cephalocaudal pattern, with eye and head control appearing before arm and torso. Furthermore, upper arm control appears before forearm control and grasp learning, and this sequence has important ramifications.
- *Coarse to fine development.* Infant abilities appear at first coarse, and are refined over time. This relates to the sensory resolution and motor control abilities, as well as to the development of skills. These embedded constraints are central to developmental growth.

We view developmental sequences as the key to skill learning, and various other works show close relevance. Grupen recognised the cephalocaudal progress of infant growth [3] and used this in skill development in robotics [4], the ITALK project has produced a robot development map [5] similar to [6], and Asada and colleagues are researching into a range of robotic models with strong emphasis on human cognitive growth [7], including the earliest stage possible; fetal development [8]. Others report on developmental approaches to reaching, including staged release [9], and experiments with proximo-distal maturation show that developmental constraints produce better learning [10].

In the following sections we will describe the development of reaching in infancy, our approach to implementing a similar sequence of development on our robotic platform, and give a series of snapshots showing the data structures built through the learning process as the robot develops reaching behaviour.

II. A DEVELOPMENTAL MODEL

In the first few months from birth infants orientate to sounds and attractive visual stimuli. They make ballistic attempts to reach towards stimulating targets but usually fail to make contact. This "pre-reaching" behaviour leads on to successful contact with objects at around 15 weeks [11], [12]. During this stage it seems that infants do not view the hand during reaching and vision is only used for target location [13]. This means that proprioception is important for arm guidance and it seems that proprioceptive development in the womb provides a more mature, although possibly incomplete, spatial framework by the time visual space is first experienced [14].

Limb movements are jerky for much this early period. The cerebellum appears to be responsible for the production of

smooth action but is very under-developed at birth. This is believed to be the cause of the marked under-damped oscillations of the arm, which gradually reduce as the cerebellum matures (over the relatively long period of 2 years).

Before 4 months there is no independent control over the fingers and grasps are formed only after contact as haptic experiences. Hand control for grasping develops later than reaching. This is an example of the cephalocaudal direction of development that is so prominent in infants [6]. It is also seen in early reaching, which involves trunk and shoulder movement, but with fingers locked. This principle of distal freezing of motor systems is an important feature and is a significant way of solving the problems associated with multiple and redundant degrees of freedom.

Only after 8 or 9 months does object size really affect approach and grasping. From this point the visually sensed object size modulates the hand aperture. Also at this age, the shift from proximal to distal control of reaching is started. It seems this is not solely due to maturational change but the trajectory of development depends heavily upon experience and patterns of behaviour [13], [15].

Another contribution to the mastery of arm control is the use of stereotypical motor patterns that have the effect of reducing the number of degrees of freedom during the early stages. By close coupling groups of muscles it is possible to reduce the number of control variables while producing a set of effective space covering actions [16]. It has also been observed that humans have a tendency to avoid extremes in arm configurations, probably because such positions considerably reduce the options for the next move. Similarly, it has been shown that people adapt their initial pose and grasp for the final arm configuration in an action task [17]. For example, subjects will choose a grasping configuration on a handle such that their hand ends up in a non-awkward position when releasing or using the object. These considerations should influence our model so that any constraining or cost function applied to reduce the DoF problem should be applied to the final configuration, not the starting configuration.

From these findings we can summarise some key points: A view of the hand is much less important when reaching than when grasping objects or other manipulations. Grasp learning follows successful reaching and involves learning object properties (affordances), finger control, tactile and other experiences. For earlier infants, who don't have much grasp control (i.e. use of fingers) proprioception may provide enough information for reaching actions.

III. EXPERIMENTAL DESIGN

We believe the developmental timeline for infants is an important tool for modeling and recognize the various constraints that pertain at different levels of development [6]. In the following sections we briefly describe how we use constraints to model the stages described in the previous section, and the resulting behaviour on the iCub. Due to space constraints the implementation details for each stage are left to the referenced papers, although the results described here are unique both in themselves and in describing the complete reach-learning process.

Following our earlier experiments, we allow eye/head coordination and eye saccading to develop independently of the construction of a proprioceptive mapping of limb space. The eye saccade learning is as described in detail in [18] and involves head movement compensation. For the growth of the proprioceptive reach space we arranged that the arm would have restricted movement on the joints for elbow and upper arm rotation, and a "rest" position was defined with the arm retracted and the hand near the head. A reach action consisted of a movement from the rest location to a specified spatial target field on or above the table surface in front of the robot. A range of target locations were generated for the volume of space around the table by motor babbling in the proprioception learning stage, (this can be done in simulation and then the locations can be transferred if motor babbling is considered unsafe on unconstrained physical hardware).

When sufficient experience has been obtained to build the gaze and reach maps the independence constraint between vision and proprioception can be lifted. This facilitates the interaction of hand and eye in behaviour known as hand regard activity. This behaviour helps by coordinating visual gaze space with the proprioceptive space of the arm/hand. Up to this point progress has been very similar to our previously described experiments [19], [20], [21].

At this stage of development the robot is able to reach to a gaze point and look at a hand position. But we notice that the gaze space is a much larger space than the reach space. This is mainly because the maximum reach is determined by the arm length which is much less than the visual range. Another important point is that the reach and gaze geometry are closely coupled in the sense that they are both grounded or referential to a point on the body centre line somewhere near the neck. This means that, regardless of the configuration of the rest of the body below the shoulders, if a stimulus is seen to be within the reachable range of the gaze/reach mappings then it can be reached. Conversely, if a stimulus is unreachable (i.e. seen but has no mapping into reach space) it can become possible to reach it by moving the head/shoulders/arm into a position where it becomes reachable. This effect can also stimulate the recruitment of locomotion to achieve distant desirable goals. However, as locomotion is not yet available, we notice that torso movement (which develops early, [11]) can be used to extend the reach space.

For the iCub, torso movements are available as tilt (forward) and rotation (about the body centre line. A torso/visual mapping can be constructed by noting the effect of torso movements on the gaze point. This process is exactly the same as the head/visual mapping which provides gaze compensation for head movements, and is described in [18]. Now, with a torso map developed, it is possible to reach to a target in a two step process: use the torso map to bring the target into a reachable location in gaze space; then use the gaze/reach mapping to generate a reach action.

The gaze space is an approximately spherical system with variables H, V and D, for left-right, up-down, and distance relative to the centre of the head. An arm configuration can be defined in terms of an n valued vector, K, for the n joint angles. Then the reach space is populated with a set of

configurations, K_i, each mapped into a gaze point, $[h_i v_i d_i]$. If motor babbling has produced a sparse but even coverage of the reach volume then we can find a K_j for an unmapped gaze point $[h_j v_j d_j]$ by interpolation between two near neighbours. Assume that $[h_1 v_1 d_1]$ and $[h_2 v_2 d_2]$ are local to $[h_j v_j d_j]$ and each are mapped, to K_1 and K_2 respectively. Then distance metrics can be computed between the vectors K_1 and K_2 and between K_1 and K_j and the resulting interpolation ratio is then applied to the elements of K_1 and K_2 to obtain a new configuration K_j on the basis of linear piecewise interpolation. If the new reach location proves to be inaccurate then its configuration can be stored, together with the mapping to $[h_i v_i d_i]$, to increase the population density of the reach space. Eventually there will be sufficient K points in the reach space that linear interpolation is effective everywhere but the space is still relatively sparse.

As described in section II, very early reaching behaviour arises before any hand control has been established and so we set the hand to be normally open with the fingers flat. If the front of the hand makes good contact with an object then an automatic finger close is executed. This provides a kind of grasp reflex which is maintained, even while the iCub performs other actions, and is only released by removal of the object, either by accident or external interaction. Unlike object contact, the release is not a significant sensed event.

As a result of the earlier hand regard behaviour the system is able to spatially correlate visual stimuli with hand positions and vice versa. Thus, when an object is presented for the first time it is likely to be detected in periphery vision and a saccade will bring the object to fixation. This fixation location in gaze space will stimulate a corresponding target for a reaching action and a reach will be initiated. At this early stage it would be expected that some reaches would miss the object and others would contact it. Some of those that make contact will also grasp the object through the grasp reflex. In accord with infant stereotypical motor patterns [22] the reach actions are completed by a return of the arm to a "home" or quiescent location in proximity to the body. (Such home positions are equivalent to the mouth, as mouthing is almost a default behaviour for any object acquired by the hand.)

After a period of early reaching, experience will have been gained on "disturbing a stimulus" (by moving it or knocking it completely out of the environment) and "holding" (with kinesthetic and possibly tactile signals). The next constraint to be lifted is the reflexive grasp and we do this by allowing the fingers to close to a given aperture and by activating a "hand empty" sensor. The hand now has potential for more control; smaller movements of the fingers can be related to visual movement or properties of objects and better grasps can be produced by matching the aperture to objects. Better approach and poise are also now within new control possibilities. Also the release of a grasped object now becomes an experienced event and so this allows objects to be dropped deliberately and thus the sophisticated skill of moving an object from one place to another is now available to learn.

In the system as described, the gaze and reach spaces record the locations of stimuli (objects) and their various properties. This is in effect a short term memory which remembers objects during saccades and reaches but a decay function ensures that after a long period without attention such recent sensory events are erased. Consequently some form of memory is required to record actions and experiences that have proved useful and can be recalled in relevant situations. We have implemented a schema learning mechanism which provides memory and motivation functions [23]. A schema encodes the context in which an action may be performed together with the result of that action. These schemas can then be chained together to carry out sequences of actions (for example, reaching toward an object, grasping it, then moving it to a new location and finally releasing it). Schemas are selected for execution based upon an intrinsic motivation algorithm which considers the novelty of currently experienced stimuli combined with their similarity to previous experiences, resulting in actions being selected which are likely to elicit new information about the world. Example schemas are shown in the next section.

IV. EXPERIMENTAL RESULTS

Following the cephalocaudal development of the infant, the robot begins by learning the eye movements required to saccade to a visual target. Learning is conducted through our developmental framework using constraints to restrict learning of sensorimotor mappings [24]. Fig.1 shows the learnt mappings between sensor and motor spaces for making eye saccades, built up by a process of motor babbling. When a stimulus is received on the retina, the mapping between the point of stimulation and the associated motor movement is followed, triggering a saccade that fixates on the stimulus.

Next, a constraint is released enabling the learning of neck control. This could be a physical constraint, such as the lack of sufficient torque in the neck, or an emergent constraint, such as the prerequisite for accurate eye saccades as a basis for learning head movements [25]. Fig. 2 shows the learnt mapping between neck muscles and the impact of these on the visual space.

The gaze space is represented by combining the motor maps from the eye and neck system. Each field in the gaze map corresponds to a relative pan and tilt movement required to fixate on that field, and contains the eye and neck movements required to do so. When performing a gaze shift to a target, the proportion of movement allocated to each system is governed by the relationship given in [26]. Although the eye and head joints are not co-located, our experiments indicate that treating them as such gives sufficient accuracy when performing gaze shifts. Depth in the gaze space is treated separately, and is calculated by the vergence angle between the two eyes.

The ego-centric gaze space shares a reference point, the torso, with the reach space. This supports the mapping of reaches to gaze direction, but also provides a space in which to represent the robot's environment. We use this space as a visual memory as well as for learning hand-eye coordination [21].

Reaching movements are mapped onto the gaze space using a combination of motor babbling and hand regard. Following the literature on early infant reaching, constraints are imposed on the type of reaches possible. In the early stages, the
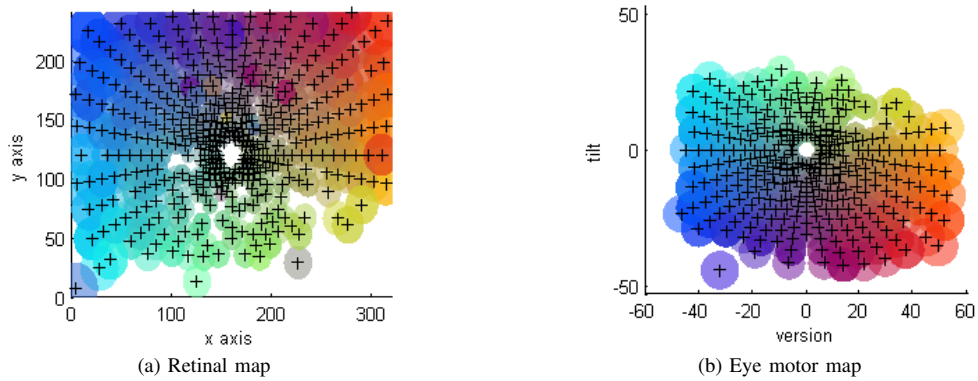
(a) Retinal map   (b) Eye motor map

Fig. 1. Maps for eye saccade control (best viewed in colour). Coloured circles indicate learnt fields over which stimuli or actions (in the case of the visual and motor maps) are considered identical. Matched colours indicate links, or mappings, between maps. A stimulus in a visual field will trigger the associated motor movement, causing the eye to saccade to the stimuli.
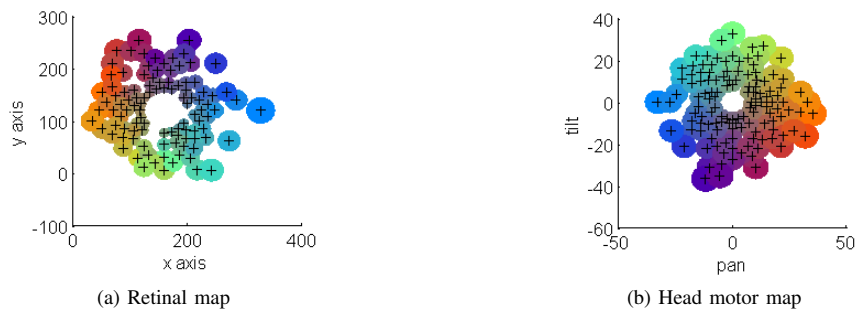

(a) Retinal map   (b) Head motor map

Fig. 2. Maps for head contributions to gaze shift. Motor movements are mapped to corresponding shifts in the visual space.

elbow joint is fixed, and "swiping" movements are made using the joints in the shoulder. Reaches are initiated from a "pre-reaching" pose with the hand near the head. This enables the robot to reach to objects on a line similar to the gaze direction, and limits collisions with other objects. The gaze-reach mapping is between two 3-dimensional spaces corresponding to shoulder proprioception and the gaze space. Fig. 3 shows a 2-dimensional projection of this mapping.

With constraints limiting elbow movement, the range of reach distances is very limited. The infant overcomes this by using movements of the torso to bring objects into range. Fig. 4 shows a mapping of torso rotation to a shift in gaze position. By rotating the torso the shoulder can be moved closer, or further, from objects to alter the distance for reaching. As the reach postures are mapped to vision through the gaze space, movement of the torso has no impact on eye-hand coordination.

At this stage, the robot is capable of gazing to objects, orientating itself to bring the objects into reaching distance, and making reaching motions toward them from the "pre-reaching" position. Using the schema learning mechanism it now starts to build composite actions from these beginnings.

When the robot sees an object it checks for schemas excited by that stimulus and finds that the most excited schema is one in which it remembers seeing its own hand in the location the object now occupies (Fig. 5a). Upon executing this the robot finds that when an object is present in the location it reaches its hand towards it receives an unexpected touch sensation.

Pre-conditions	Action	Post-conditions
	Reach to 35,-66	Hand at 35,-66

(a) Initially excited schema

Pre-conditions	Action	Post-conditions
Obj. 1 at 35,-66	Reach to 35,-66	Obj. 1 at 35,-66 Hand at 35,-66 Touching obj. 1

(b) Extended schema with new information

Pre-conditions	Action	Post-conditions
Obj. $a at $x,$y	Reach to $x,$y	Obj. $a at $x,$y Hand at $x,$y Touching obj. $a

(c) Generalised schema

Fig. 5. The creation of a schema representing the act of touching objects

A new schema is then formed to represent this knowledge (Fig. 5b), which can then be generalised in to a form which represents reaching out and touching objects in any position (Fig. 5c).

The new touching schema is executed a number of times due to the novelty of the experiences involved. However after a short while the excitation drops below that of the next most excited schema, which in this case is the grasping schema. The grasping schema is excited by the memory of the robot touching its own hand when performing a grasp with no objects present, which it is reminded of by the touch sensation it receives from the object it has reached towards (Fig. 6a).
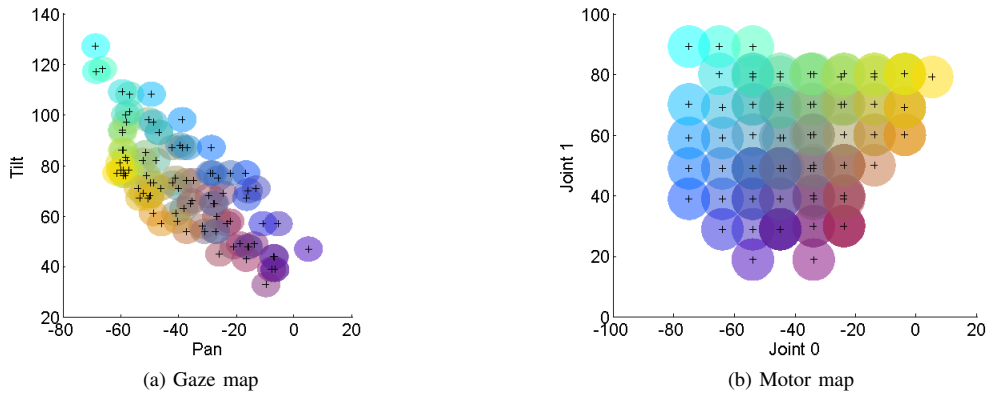
(a) Gaze map

(b) Motor map

Fig. 3.   2 dimensional projection of reach maps in gaze and motor space

(a) Gaze map

(b) Torso motor map
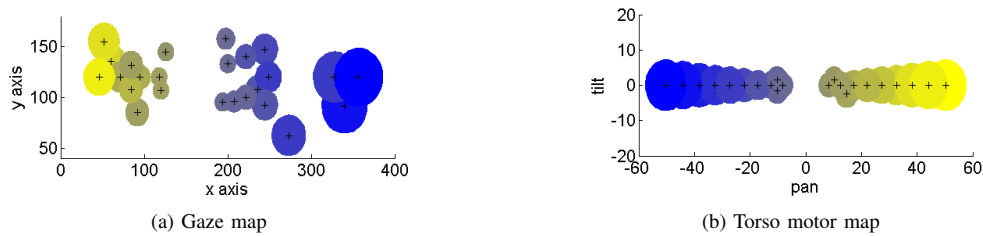
Fig. 4.   Maps for torso rotation impact on the gaze direction

Pre-conditions	Action	Post-conditions
	Grasp	Touching hand

(a) Initially excited schema

Pre-conditions	Action	Post-conditions
Obj. 1 at 35,-66 Touching obj. 1	Grasp	Obj. 1 at 35,-66 Hand at 35,-66 Holding obj. 1

(b) Extended schema with new information

Pre-conditions	Action	Post-conditions
Obj. $a at $x,$y Touching obj. $a	Grasp	Obj. $a at $x,$y Hand at $x,$y Holding obj. $a

(c) Generalised schema

Fig. 6.   The schema memory learns to go from touching to grasping objects

Pre-conditions	Action	Post-conditions
Obj. $a at $x,$y	Reach to $x,$y	Obj. $a at $x,$y Hand at $x,$y Touching obj. $a

Pre-conditions	Action	Post-conditions
Obj. $a at $x,$y Touching obj. $a	Grasp	Obj. $a at $x,$y Hand at $x,$y Holding obj. $a

Fig. 7.   Chaining of touching and grasping schemas

Executing this whilst touching an object results in the robot successfully grasping the object and receiving the sensation of holding an object. A new schema is then created to represent this new information (Fig. 6b). As with the new touching schema this grasping representation can also be generalised as shown in Fig. 6c, which represents the act of grasping an object in any location.

In the last stage of our current implementation the touching and grasping schemas can be chained together to form a plan of action which allows the robot to reach towards and then grasp an object at any location (Fig. 7).

This completes the process of attaining visually elicited reaching. Learning is driven by novelty in the early stages, giving way to goal directed behaviour only when suitable goals have been found through 'play'. The sequence shows cumulative learning of skills from sensorimotor mapping to action planning. A key indication of the power of this approach is that the whole sequence described here can be run on the iCub robot in just 2.5 hours.

A critical issue is the scheduling of the release of constraints. In connected work we have investigated how the timing of constraint release impacts on learning of gaze control [18]. Those results showed a trade off between timing of constraint release and the rate of learning. If there are no sequencing constraints, then sub-systems are allowed to learn in parallel and learning is found to be slow, due to added physical and computational complexity. Correspondingly, connectivity between maps is sparse. If constraints remain in place for a prolonged period, learning of the unconstrained system is initially fast and connectivity is high, but at the expense of improvement in the constrained system. However, learning saturates as the space becomes increasingly explored. By releasing the constraint on a sub-system at an intermediate

time, learning of mappings in both systems is increased. Preliminary results suggest that the optimal time to release a constraint to maximise learning depends on the interaction of the codependent learning rates of the systems involved. This is a matter for further investigation.

V. CONCLUSIONS

We have described the nature of development of reaching in human infancy, and how stages in the development provide useful insights for learning to reach in humanoid robotics. We have taken these ideas and implemented them on a robotic platform using our framework for developmental learning. Results show snapshots of the sensorimotor mappings and schemas learnt along the developmental trajectory in a cephalocaudal manner from making eye saccades to reaching and grasping. The work shows the value of several principles we draw from the developmental literature.

Motor babbling is a key element in learning. The limited abilities of the infant mean that goal-driven learning is absent or restricted, and intrinsic activity, in the form of motor babbling, plays a significant role in early development. But babbling, which has close links to play behaviour, is more than random behaviour, generating vital sensorimotor data and rehearsing prior action and experience.

Proprioceptive space is an important and under-rated perceptual substrate in early learning. Before vision has developed sufficiently, proprioception provides the main feedback on limb positioning. This allows limb movements to be learnt, to some extent, prenatally. Once vision has matured, motions learnt proprioceptively can be refined with visual feedback.

Certain abilities, such as gaze control, must be refined before others, such as reaching. This is manifest in the cephalocaudal sequence of development. Furthermore, constraining distal joints until control over proximal ones has been learnt, structures the learning task. In this case, by restricting motion at the elbow joint the robot is able to learn shoulder control with a straightforward mapping.

The resolution of sensor and motor abilities in the infant are initially coarse, and gradually become finer with neural development and learning. Viewing this phenomena in terms of constraints allows us to reflect the developmental trajectory in robotic systems. As the robot masters coarse abilities, relevant constraints can be lifted to allow further refinement.

Using these lessons from human development we have built a robotic system that learns to reach in a way that overcomes many of the hurdles to humanoid reaching. Our experiments can be seen to expose some of the "logic" that appears to be behind the infant's development in early sensory-motor learning. We believe this continued approach will offer further valuable models and solutions for robotics.

REFERENCES

[1] O. Sporns and G. Edelman, "Solving Bernstein's Problem: A Proposal for the Development of Coordinated Movement by Selection," *Child Development*, vol. 64, no. 4, pp. 960–981, 1993.

[2] F. Nori, L. Natale, G. Sandini, and G. Metta, "Autonomous learning of 3D reaching in a humanoid robot," in *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*. IEEE, 2007, pp. 1142–1147.

[3] R. Grupen, "A developmental organization for robot behavior," Dept Of Computer Science, Massachusetts University, Amherst., DTIC Document ADA439115, 2005.

[4] J. Coelho, J. Piater, and R. Grupen, "Developing haptic and visual perceptual categories for reaching and grasping with a humanoid robot," *Robotics and Autonomous Systems*, vol. 37, no. 2, pp. 195–218, 2001.

[5] A. Cangelosi, G. Metta, G. Sagerer, S. Nolfi, C. Nehaniv, K. Fischer, J. Tani, T. Belpaeme, G. Sandini, F. Nori *et al.*, "Integration of action and language knowledge: A roadmap for developmental robotics," *Autonomous Mental Development, IEEE Transactions on*, vol. 2, no. 3, pp. 167–195, 2010.

[6] J. Law, M. Lee, M. Hülse, and A. Tomassetti, "The infant development timeline and its application to robot shaping," *Adaptive Behaviour*, vol. 19, pp. 335–358, 2011.

[7] M. Asada, K. Hosoda, Y. Kuniyoshi, H. Ishiguro, T. Inui, Y. Yoshikawa, M. Ogino, and C. Yoshida, "Cognitive developmental robotics: a survey," *Autonomous Mental Development, IEEE Transactions on*, vol. 1, no. 1, pp. 12–34, 2009.

[8] H. Mori and Y. Kuniyoshi, "A human fetus development simulation: Self-organization of behaviors through tactile sensation," in *Development and Learning (ICDL), 2010 IEEE 9th International Conference on*. IEEE, 2010, pp. 82–87.

[9] P. Savastano and S. Nolfi, "Incremental learning in a 14 dof simulated icub robot: Modeling infant reach/grasp development," *Biomimetic and Biohybrid Systems*, pp. 250–261, 2012.

[10] S. Ramırez-Contla, A. Cangelosi, and D. Marocco, "Developing motor skills for reaching by progressively unlocking degrees of freedom on the icub humanoid robot," in *Proceedings of the Post-Graduate Conference on Robotics and Development of Cognition*, 2012.

[11] M. R. Fiorentino, *A Basis for Sensorimotor Development, Normal and Abnormal: The Influence of Primitive, Postural Reflexes on the Development and Distribution of Tone*. Springfield, Ill: Charles C Thomas, 1981.

[12] C. Von Hofsten, "Prospective control: A basic aspect of action development," *Human Development*, vol. 36, pp. 253–270, Sep. 1993.

[13] N. E. Berthier and R. L. Carrico, "Visual information and object size in infant reaching," *Infant Behavior and Development*, vol. 33, no. 4, pp. 555 – 566, 2010.

[14] B. Hopkins, *Prenatal development of postnatal functions*. Praeger Pub Text, 2005.

[15] C. von Hofsten and L. Rönnqvist, "Preparation for grasping an object: A developmental study," *Journal of Experimental Psychology: Human Perception and Performance*, vol. 14, pp. 610–621, 1988.

[16] M. Latash, *Synergy*. Oxford University Press, USA, 2008.

[17] D. Rosenbaum, E. Halloran, and R. Cohen, "Grasping movement plans," *Psychonomic bulletin & review*, vol. 13, no. 5, pp. 918–922, 2006.

[18] J. Law, P. Shaw, and M. Lee, "A biologically constrained architecture for developmental learning of eye-head gaze control on a humanoid robot," *Autonomous Robots*, Under review.

[19] M. H. Lee, Q. Meng, and F. Chao, "Developmental learning for autonomous robots," *Robotics and Autonomous Systems*, vol. 55, no. 9, pp. 750–759, 2007.

[20] F. Chao, M. Lee, and J. Lee, "A developmental algorithm for ocularmotor coordination," *Robotics and Autonomous Systems*, vol. 58, pp. 239–248, 2010.

[21] M. Hülse, S. McBride, J. Law, and M. Lee, "Integration of active vision and reaching from a developmental robotics perspective," *IEEE Transactions on Autonomous Mental Development*, vol. 4, no. 2, pp. 355–367, 2010.

[22] N. E. Berthier, R. K. Clifton, D. D. McCall, and D. J. Robin, "Proximodistal structure of early reaching in human infants," *Experimental Brain Research*, vol. 127, pp. 259–269, 1999.

[23] M. T. Sheldon and M. Lee, "PSchema: A developmental schema learning framework for embodied agents," in *Proceedings of the IEEE Joint International Conference on Development and Learning, and Epigenetic Robotics*. Frankfurt, Germany: IEEE, 2011.

[24] M. Lee, Q. Meng, and F. Chao, "Staged competence learning in developmental robotics," *Adaptive Behaviour*, vol. 15, no. 3, pp. 241–255, 2007.

[25] P. Shaw, J. Law, and M. Lee, "A comparison of learning strategies for biologically constrained development of gaze control on an icub robot," *Autonomous Robots*, Under review.

[26] X. Wang and J. Jin, "A quantitive analysis for decomposing visual signal of the gaze displacement," in *Proceedings of the Pan-Sydney area workshop on Visual information processing - Volume 11 (VIP '01)*, D. D. Feng, J. Jin, P. Eades, and H. Yan, Eds., vol. 11. Darlinghurst, Australia: Australian Computer Society, Inc., 2001, pp. 153–159.

A.4 ROSSI Project: Internal memo on peripersonal vs. extrapersonal object interaction

A robotic perspective on peripersonal and extrapersonal object interaction

Michael Sheldon, Mark Lee

October 3, 2011

Abstract

In this paper we investigate a robotic implementation of a set of experiments previously performed on human participants. These experiments were part of a psychological study looking at the effects of peripersonal and extrapersonal space on reactions to different classes of verbs. We then present two hypotheses based upon our robotic tests that may relate to the processes occurring within the human brain.

1 Introduction

Ambrosini, et al. previously investigated the effect of the spatial position of objects in relation to verbs [1]. They found that when subjects heard a word relating to functional use or manipulation of an object that was within the subject's own peripersonal space that reaction times would be shorter than when hearing similar verbs paired with an object that was in the subject's extrapersonal space. In contrast, words relating to observation of or pointing at an object resulted in similar reaction times, regardless of whether the object was in the peripersonal or extrapersonal space of the subject.

In this paper we attempt to reproduce this in a robotic setting, making use of a developmental schema learning framework that we have been using to investigate potential methods for the emergence of communication in robots, following a similar developmental progression to that found in human infants [11]. Based upon our robotic results we then suggest two potential hypotheses to explain the differences in reaction time found in the previous human experiments.

2 Related Research

Pointing in our system emerges as a form of *proto-imperative pointing* resulting out of failed grasping behaviour [13, 7]. Proto-imperative pointing is used by a child to indicate an object of desire to a nearby adult, and typically emerges at around 10-12 months. On average 3 months [2] after the emergence of proto-imperative pointing the child has also learnt to perform *proto-declarative pointing* which is used to acquire joint attention on an object with an adult. Our system does not make this leap to proto-declarative pointing, indeed Masataka suggests that proto-declarative pointing does not emerge from the same developmental progression as proto-imperative pointing [9] and Povinelli, et al. [10] show that while chimpanzees raised in captivity can be trained to perform proto-imperative pointing they do not appear to make the jump to proto-declarative pointing. As such all pointing performed in the following experiments is proto-imperative in nature, with the robot expecting its pointing actions to trigger assistance from nearby humans.

Lee, et al. [6, 5] discuss the use of a Lift Constraint, Act, Saturate (LCAS) 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. This technique is made use of in the training of our robot prior to the experimentation stage. Constraints are placed upon the system's sensory input and the system then operates in this mode until there is little novel 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.

The developmental progression followed by the system can be seen in section 3 and is inspired by the work of Iverson and Goldin-Meadow [4], highlighting pointing as a key stage towards more complex forms of communication.

The learning within our system is supported by a schema learning framework [12, 3] capable of forming generalised schemas representing different concepts and associating additional sensory information (such as the hearing of words) alongside these schemas.

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. Schemas are selected for execution based upon an excitation measure favouring novelty of experience [12].

3 Experimental Method

We first trained a schema memory on a simulated robot within the Gazebo simulator, which provides a 3D environment with rigid body physics simulation. This can be seen in figure 1. The simulated robot consists of a robot arm mounted on a vertical backplane. 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 for communicating by pointing at an object. The simulation also includes a camera looking down on the table from above the arm in a similar configuration to that of a human head and arms, the vision system is able to recognise the robot's own end effector and objects placed on the table.

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, an illustration of these fields can be seen in figure 2.

The robot has two fundamental types of action it can perform, joint movements and observations. Joint movements allow the system to specify a joint configuration that the arm will then move to, it is out of this type of action that touching and pointing later emerge. Observation actions cause the robot to focus visually on one field and list the contents of that field (in addition to any sensory information being received normally). These observations are indicated visually by displaying an enlarged view of that field, as can be seen in figure 2.

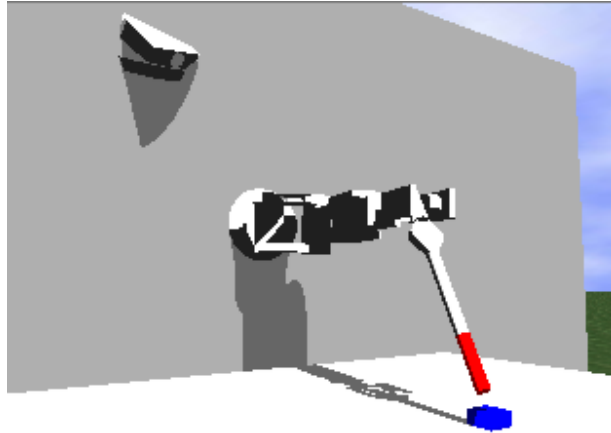Figure 1: The simulation environment showing the arm pointing at an object placed slightly outside of the robot's work envelope.
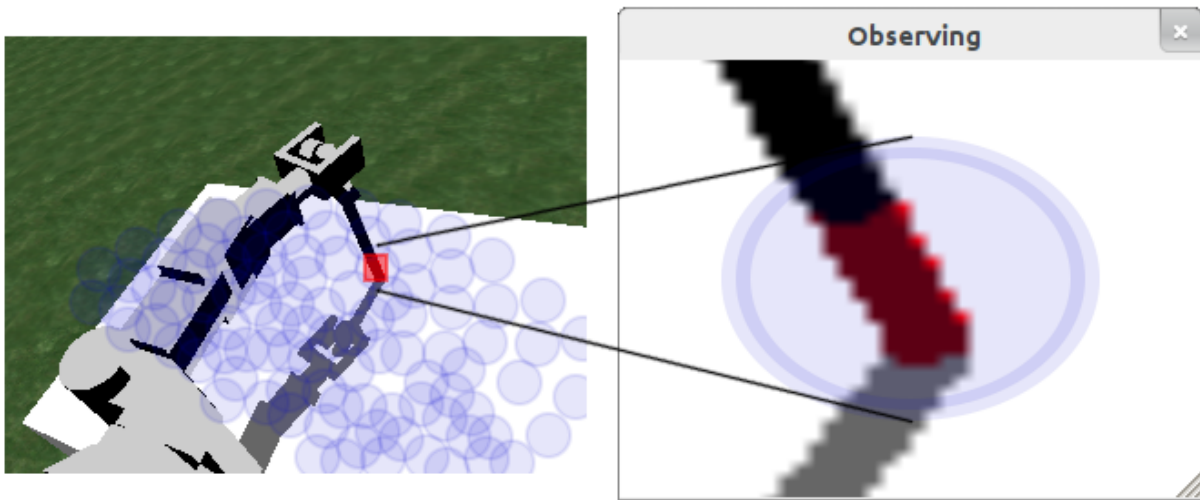

Figure 2: *Left:* The view from the robot's camera with a visualisation of the visual fields superimposed and detected objects highlighted (in this case the robot's own end-effector). *Right:* An example of the robot observing a field containing its own end effector.

The memory is trained following a developmental progression consisting of the following stages:

3.1 Motor babbling

In this initial stage the robot has had no prior experience of the world or of its own body. It performs spontaneous motor actions in order to discover the properties of its motor systems and its anatomical constraints.

3.2 Motor vision mapping

The movements learnt in the previous stage are then mapped to the changes they create in the robot's vision system, this allows it to move its arm to touch (or point towards) an object detected visually.

3.3 Object interaction

The robot is then presented with objects in a number of different positions and the excitation mechanism guides the robot towards previously performed actions that may be relevant to this new stimulus. This results in the robot moving its finger into the same field as the object and receiving a touch sensation. After seeing objects in a few different locations it is able to generalise this into a schema that represents the concept of touching anything in any position.

3.4 Failed grasping leading to pointing

In attempting to touch objects that lie outside of the work-envelope of the robot it will incidentally perform what looks, to a human observer, like a pointing motion. Through assistance from a human observer, fetching the indicated object for the robot, the robot's representation of this action moves away from being a direct attempt at manipulating the world towards an indirect effect via external agencies. This could be seen as a very early form of social communication being incorporated into the robot's experience.

3.5 Verb learning

The robot is then given auditory input (reduced to a text token by speech recognition software) whilst it performs actions. This input is directly related to the action being taken, the words used being *'touch'*, *'point'* and *'observe'*.

This progression is discussed in more detail in [11], while the schema learning mechanism used to support this progression is described in [12].

The system is trained to a level whereby it can understand the verbs *'touch'*, *'observe'* and *'point'* and can relate them to relevant concepts within its memory. We then transplant this schema memory into a test framework that provides pre-determined pairs of stimuli consisting of visual observations and verbs, similar to the pairs used by Ambrosini, et al. in [1]. Having no capabilities relating to the function of objects this class of verbs is not covered in our experiments.

An object which is visually distinct from those that the system had previously been trained on is presented, either inside the peripersonal space or inside the extrapersonal space. Along side this visual stimulus a word is heard, either *'touch'*, *'observe'* or *'point'*. We then record the schemas or chains of schemas most strongly activated by each pairing.

4

4   Results

We presented the system with visual information relating to an object and its position paired with one of the three verbs previously listed. We then looked at the schemas or chains of schemas selected for each position/verb combination.

In each of the examples below the schema displayed is the final form of the schema as determined by the schema learning framework. In each case they start off as more general schemas which are then made into schemas specific to the current world state based on the system's current observations of the world.

Figure 3 shows the result of the system being presented with an object in the **peripersonal** space and hearing the word *'observe'*, an observation schema relating to the field the object is in is then selected. In this schema the system expects that if it focuses its attention on field 23 (the field containing the object) it will then experience both the sensation of observing that field and of observing the object contained within.

Pre-conditions	Action	Post-conditions
Object 5 in field 23	Observe field 23	Object in field 23
		Observing field 23
		Observing object 5

Figure 3: Object is placed in peripersonal space and the word *'observe'* is heard.

Figure 4 shows the result of the system being presented with an object in the **extrapersonal** space and hearing the word *'observe*. As with the previous pairing an observation schema relating to the field containing the object is selected, showing that no distinction is made between peripersonal and extrapersonal space when performing observation actions.

Pre-conditions	Action	Post-conditions
Object 5 in field 87	Observe field 87	Object in field 87
		Observing field 87
		Observing object 5

Figure 4: Object is placed in extrapersonal space and the word *'observe'* is heard.

Figure 5 shows the result of the system being presented with an object in the **extrapersonal** space and hearing the word *'point'* a typical pointing schema is selected. The system selects a schema which would result in it pointing at the object with the expectation that another agent may move it into a position closer to the robot (in this case field 54, as this is where the object has been placed most frequently during the earlier training phase). The robot has this expectation because it has learnt to point proto-imperatively, and so views this gesture as being a request for the object to be moved.

Pre-conditions	Action	Post-conditions
Object 5 in field 87	Move to joint configuration 1.39, 0.00	Object in field 54
		Finger in field 87

Figure 5: Object is placed in extrapersonal space and the word *'point'* is heard.

Figure 6 shows the result of the system being presented with an object in the **peripersonal** space and hearing the word *'touch'*. The system selects a schema appropriate to touching the object with the expectation that it will receive a touch sensation after moving to a joint configuration that will result in its finger being in the same visual position as the object.

Pre-conditions	Action	Post-conditions
Object in field 32	Move to joint configuration 0.87, 2.26	Object in field 32 Finger in field 32 Touching

Figure 6: Object is placed in peripersonal space and the word *'touch'* is heard.

Figure 7 shows the result of the system being presented with an object in the **extrapersonal** space and hearing the word *'touch'*. In this case the most relevant schema to the given stimuli is found to be impossible to execute in the current world state due to its pre-conditions (of the object being reachable) not being met. The system resolves this by finding a chain of actions which will make it possible to achieve the desired goal of touching the object. It does this by first selecting a pointing schema which it believes will result in the object being moved into a location in which it can then be touched. This belief is due to the robot having learnt to point proto-imperatively, using pointing as a requesting action for an object which another agent may then fetch for it.

Pre-conditions	Action	Post-conditions
Object 5 in field 87	Move to joint configuration 1.39, 0.00	Object in field 54 Finger in field 87

Pre-conditions	Action	Post-conditions
Object in field 54	Move to joint configuration 1.75, 1.92	Object in field 54 Finger in field 54 Touching

Figure 7: Object is place in extrapersonal space and the word *'touch'* is heard.

5   Conclusions

The results above suggest two potential hypotheses for the differences in reaction time found by Ambrosini, et al. We do not believe these two hypotheses to necessarily be mutually exclusive.

5.1   Hypothesis I: Planning/Simulation

The selected schema plan shown in figure 7 indicates that when the system is presented with an object that is in its extrapersonal space and is asked to touch that object it forms a chain of schemas which would allow it to perform the requested action. It is possible that the increased reaction time seen in the human experiments when presenting an object in extrapersonal space paired with a functional or manipulative verb is caused by the brain evaluating possible courses of action that may resolve the problem of it being out of reach.

While adult humans may not typically tend to point proto-imperatively as our robot does, they do have a range of other potential actions that serve the same purpose (requesting the object verbally, moving their body closer to the object, using other objects as tools to reach the target object, etc.) which could be being partially activated as part of a planning/simulation process within the brain.

5.2 Hypothesis II: Past Experience

The robot has prior experience of having successfully touched objects which are within its peripersonal space as well as experience of having failed to touch objects which are in its extrapersonal space (as would a human). When considered from a connectionist perspective it could be suggested that this might lead to a stronger/more direct neural connection between the areas of the motor-vision map which represent the peripersonal space with the concept of touching or manipulating an object, than between the areas representing extrapersonal space and the touching or manipulating of an object.

However the results from Ambrosini, et al. indicate no significant difference in reaction time between pointing in peripersonal and extrapersonal space, while pointing is a gesture more strongly associated with distant objects (although not exclusively). This may suggest that there is more going on than simply differences in the strength of neural connections based on prior experiences.

6 Further Work

It may be possible to strengthen or rule out the case for hypothesis I by reproducing the original human experiments whilst making use of brain imaging techniques to identify any increased activity related to planning when hearing functional or manipulative verbs alongside objects in the extrapersonal space as opposed to those same verbs heard with an object in the peripersonal space.

References

[1] E. Ambrosini, Scorolli C., A. M. Borghi, and M. Costantini. Which Body for Embodied Cognition. *To be published*, 2011.

[2] L. Camaioni, P. Perucchini, F. Muratori, and A. Milone. Brief report: A longitudinal examination of the communicative gestures deficit in young children with autism. *Journal of autism and developmental disorders*, 27(6):715–725, 1997.

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

[4] J. M. Iverson and S. Goldin-Meadow. Gesture Paves the Way for Language Development. *Psychological Science*, 16(5):367–371, 2005.

[5] M. H. Lee, Q. Meng, and F. Chao. Developmental learning for autonomous robots. *Robotics and Autonomous Systems*, 55(9):750–759, 2007.

[6] M. H. Lee, Q. Meng, and F. Chao. Staged competence learning in developmental robotics. *Adaptive Behavior*, 15(3):241, 2007.

[7] J. Lindblom and T. Ziemke. Social Situatedness of Natural and Artificial Intelligence: Vygotsky and Beyond. *Adaptive Behavior*, 11(2):79, 2003.

[8] M. Lungarella, G. Metta, R. Pfeifer, and G. Sandini. Developmental robotics: a survey. *Connection Science*, 15(4):151–190, 2003.

[9] N. Masataka. From index-finger extension to index-finger pointing: Ontogenesis of pointing in preverbal infants. *Pointing: Where language, culture, and cognition meet*, pages 69–84, 2003.

[10] D. J. Povinelli, J. E. Reaux, D. T. Bierschwale, A. D. Allain, and B. B. Simon. Exploitation of pointing as a referential gesture in young children, but not adolescent chimpanzees. *Cognitive Development*, 12(4):423–461, 1997.

[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] M. Sheldon and M. Lee. PSchema: A developmental schema learning framework for embodied agents. In *Proceedings of the First Annual IEEE Joint International Conference on Development and Learning and Epigenetic Robotics*. IEEE, 2011.

[13] L. S. Vygotsky. Mind In Society. page 56, 1978.