Developmental reasoning and planning with robot through enactive interaction with human

Maxime Petit

N°d'ordre 37 - 2014 Année 2013

THESE DE L'UNIVERSITE DE LYON

présentée devant

L'UNIVERSITE CLAUDE BERNARD LYON 1

Ecole Doctorale Neurosciences et Cognition

pour l'obtention du

Diplôme de Thèse

(arrêté du 7 août 2006)

Discipline : SCIENCES COGNITIVES

Option : INFORMATIQUE

présentée et soutenue publiquement le 6 Mars 2014

par

Maxime Petit

Raisonnement et Planification Développementale d'un Robot via une Interaction Enactive avec un Humain

Developmental Reasoning and Planning with Robot through Enactive Interaction with Human

---

dirigée par Peter F. Dominey

devant le jury composé de :

Pr. Rémi Gervais	Président du jury
Pr. Giorgio Metta	Rapporteur
Pr. Philippe Gaussier	Rapporteur
Pr. Chrystopher Nehaniv	Examinateur
Dr. Jean-Christophe Baillie	Examinateur
Dr. Peter Ford Dominey	Directeur de thèse

Stem-Cell and Brain Research Institute, INSERM U846
18, avenue Doyen Lepine
6975 Bron Cedex

École doctorale Neurosciences et Cognition (ED 476 - NSCo)
UCBL - Lyon 1 - Campus de Gerland
50, avenue Tony Garnier 69366 Lyon Cedex 07

Pour mon père. Qu'il puisse être fier : tu vois, j'y suis arrivé finalement. Cette thèse est pour toi.

Pratchett, "Science of the Discworld II : the Globe" :

'An important part of the Make-a-Human kit is the Story. We tell our children stories, and through those stories they learn what it is like to be a member of our tribe or our culture. [...]. We use stories to build our brain, and then we use the brains to tell ourselves, and each other, stories.'

'We had to find a way to share our intelligence with others, and to store useful ideas and tricks for the benefit of the whole group, or at least, those in position to make use of it. That's where extelligence comes into play. Extelligence is what really gave those apes the springboard that would launch them into sentience, civilisation, technology, and all the other things to make humans unique on this planet. Extelligence amplifies the individual's ability to do good - or evil. It even creates new forms of good and evil, such as, respectively, cooperation and war.'

Remerciements

Je tiens à remercier Monsieur Peter Ford Dominey, Directeur de Recherche CNRS à l'INSERM U846, pour m'avoir donné la possibilité de travailler dans le domaine de la robotique intelligente et d'avoir encadré cette thèse, ainsi que pour son aide et tous ses conseils pendant toute sa durée.

Je remercie également Monsieur Stéphane Lallée, Docteur en Robotique et mon prédecesseur : *"Always two there are, a master and an apprentice"*. Merci pour m'avoir laissé une épaule par dessus laquelle regarder et tous les réponses à mes problèmes théoriques et pratiques que j'ai pu avoir.

J'aimerais adresser un remerciement particulier à Monsieur Grégoire Pointeau, doctorant, qui m'a suivi dans cette aventure. C'est avec ta collaboration que cette thèse a pu avoir lieu, que cela soit au niveau scientifique ou de l'ambiance de travail : merci pour ton humour... particulier, et vive les Pêcheurs de Saumons! :D

Merci également à Monsieur Guillaume Gibert, Post-Doctorant et Monsieur Florian Lance, Ingénieur, pour leurs aides techniques avec les diverses réparations du robot (neurochirurgie et chirurgie orthopédique) ou les discussions scientifiques dans non pas le Roundworld mais dans l'Icubworld.

Merci aussi à tous l'équipe Integrative Neurosciences and Robotics pour les différents échanges, journal club, discussion et soutien : Madame Jocelyne Ventre-Dominey, Madame Carol Madden, Monsieur Sullivan Hidot, Monsieur Xavier Hinaut, Monsieur Pierre Enel.

J'en profite aussi pour remercier toutes les personnes du laboratoire que j'ai pu connaître, et qui ont su accepter un Geek au sein de leur environnement biologique. Je vais en oublier et me faire taper, mais en particulier Sophie (Super Marraine!), Julie F., Cédric, Julie W., Angèle, Diana, Anne-Lise, Loïc, Pierre, Pierrot, Yann,... Les différents "journal club" et autre présentation de mémoire ou de thèse, les conversations scientifiques ou non pendant les pauses cafés, déjeuner ou les soirées ont participé à tout cela, en particulier en m'apportant de la lumière naturelle et de la bonne humeur. En espérant aussi avoir pu laisser en retour un petit quelque chose : un kruskall-wallis, un masque de catcheur, un RTFMN ou un "Force et Honneur!".

A présent, je tiens à remercier toute ma famille qui m'a supporté, pas seulement depuis le début de la thèse mais aussi pour tout le chemin qui m'y a mené. Mes parents évidemment, qui ont toujours été derrière moi et m'ont permis d'étudier dans les meilleures conditions. Ma soeur Adeline et mon frère Renaud pour me sortir un peu du domaine scientifique et de la recherche, Sébastien pour ses conseils et ses réparations express (c'est toi le vrai geek!). Enfin merci à Maël et Nayann : vos sourires lorsque je rentrais vous

voir me reboostaient pour un mois entier. Merci aussi à Nicole et Christiane et mes autres tantes et oncles, cousins, cousines et toute ma famille : que cela soit à Châlons, à Reims ou à Plouharnel, vous étiez toujours là pour m'autoriser un moment de détente, avant de revenir à Lyon sous de meilleurs conditions.

Merci aussi à Rita, pour avoir su gratter sous la couche du Geek et m'accepter dans sa vie. Ton soutien inamovible (malgré mes deux mains gauches!), tes petits plats et tes attentions et ta patience m'ont permis de réaliser cette thèse. Obrigado minha Ritinha

Enfin merci à tous mes amis, pour leurs conseils, leurs invitations et tous les moments partagés au cours de cette aventure : Nicolas, Jérémi, Clément, Benoît à Châlons (pendant le traditionnel dîner de Noël), Richard et Juliana à Paris (et aux autres Bidons!), Céline et Thomas à Vannes (POUTOUX à François et Maxime), Luc à Montpellier (j'ai essayé d'éviter le Dark Side), Seb et Flo à Marseille (et les adorables Cyrielle et Vivanne), Nicolas et Karine à Lausanne (INSA power!), Camille à Lille (INSA aussi i, et Hugo (qui m'a supporté en tant que collocataire), Laurent (merci pour les magnifiques parties de JDR et de FIFA), Mattéo, Louis et Grégoire de Lyon (et de Metal Adventure, Capharnaüm ou autre univers fantastique).

Abstract

Résumé

Que cela soit par des automates puis par des robots, l'Homme a été fasciné par des machines pouvant exécuter des tâches pour lui, dans de nombreux domaines, comme l'industrie ou les services : c'est ce dernier domaine qui nous sert de contexte.

Ainsi, nous avons utilisé une approche développementale, où le robot se doit d'apprendre de nouvelles tâches au cours de sa vie. Inspiré par des théories sur le développement de l'enfant, nous avons extrait les concepts intéressants pour les implémenter sur une plateforme robotique humanoïde : l'iCub.

L'acquisition du langage est une première étape, où la capacité à classifier les mots de classes ouvertes et de classes fermées permet d'obtenir une syntaxe qui aide l'enfant à construire le lien entre une phrase et son sens. Cette méthode a été implémentée grâce à un réseau de neurones récurrents, utilisant une base de données fournit par l'humain en interagissant avec le robot.

La maîtrise du langage permet à l'enfant de participer à des actions plus complexes, en particulier des tâches collaboratives où la parole est requise de négocier le mode d'apprentissage sur plusieurs modalités. Implémenté sur l'iCub et le Nao, cela permet un apprentissage en temps réel et de réaliser un plan partagé.

Enfin, nous avons étudié le fonctionnement de la mémoire autobiographique, cruciale pour se remémorer des épisodes passés de sa vie, d'en tirer des prédictions et de les appliquer dans le futur. En recréant cette mémoire en SQL et formatant les données en PDDL, l'iCub est alors capable de raisonner en fonction de sa propre expérience, lui permettant ainsi de résoudre le problème des Tours d'Hanoi sans jamais l'avoir visualisé avant.

Mots-clefs

Robots, iCub, Nao, Robotique Développementale, Cognition, Interaction Homme-Robot et Coopération, Langage, Apprentissage situé et social

Developmental Reasoning and Planning with Robot through Enactive Interaction with Human

Abstract

From automata to robots, the Human has always been fascinated by machines which could execute tasks for him, in several domains like industry or services.

Indeed, we have used a developmental approach, where the robot has to learn new tasks during his life. Inspired by theories in child development, we have extracted the interesting concepts to implement them on a humanoid robotic platform : the iCub.

Language acquisition is a first step, where the capacity to classify closed and opened class words allows to obtain a syntax which help the children to make the link between a sentence and its meaning. This method has been implemented with a recurrent neural network, using a database provided from the human by interaction with the robot.

The control of the language allows the children to participate in more complex actions, in particular cooperative tasks, where speech is required to negotiate the learning mode within several modalities. Implemented on the iCub and the Nao, this allows a real-time learning and to realize a shared plan.

Eventually, we have studied the functioning of the autobiographical memory, crucial to remember episodes of his life, to extract predictions from and to apply them in the future. By recreating this memory in SQL, and by formatting the data in PDDL, the iCub is then capable of reasoning in function of his own experience, allowing him to solve the Tower of Hanoi problem without knowing the solution before.

Keywords

Robots, iCub, Nao, Developmental Robotics, Cognition, Human-Robot Interaction and Cooperation, Language, Situated and Social Learning

Contents

Introduction

Robots. This thesis is about robots. But, what is a robot?

Everyone could picture a robot in his mind, some of them are real (Roomba, Nao, ...), some come from science-fiction (R2D2, Wall-E, ...), some may even be coming from their imagination. But overall, it is most likely that two people will think about two different robots. As the topic of this thesis, it is important to define and describe what a robot is. It is such a basic question, but in fact a very tricky one : so difficult than the definition of robot is controversial, including among scientists inside the very same field.

In fact, it is particularly difficult to have a proper and universal definition because the ethymology of the word does not come from an old language (e.g. greek or latin, base for the majority of the name of scientific disciplines) but is quite new and take his roots in science-fiction novels.

Indeed, the first appearance of the word "robot" was used by a Czech writer, Karel Čapek [1], in 1920 in a play called *Rossom's Universal Robots* (more commonly named R.U.R.), to describe artificial workers, as shown in Figure 1 (from Czech, *robotna* : compulsory labor).

If we look at an actual dictionnary, Oxford's explain that it is *"a machine capable of carrying out a complex series of actions automatically"* whereas Merriam-Webster's definition is *"a device that automatically performs complicated often repetitive tasks"*. These definitions are general (but not general enough as we will see in the next section) and formed around the "automatically" property of a robot. In fact, one of the father of robotics, Engelberger has said *"I can't define a robot, but I know one when I see one"*. One explanation for the origin of this issue is that the word has first been invented, and then the field and the ideas have come to life. But in fact, if the robotic field has received his "name" from Čapek less than 100 years ago, it has emerged and matured over several millenia.

In the chapter 1, starting from these definitions, we will review the robotics history, in order to better understand the field, to list the properties needed to be called "robot" and to identify the problem that were faced and sometimes resolved until now. Indeed, we will go from the ancestor of the robot, the automata, built and developed from the Antiquity, to the modern robotics, with a special focus on the service robots which is our particular interest. The intrinsic difficulties encountered by the robot in this precise use will lead us to take the developmental approach which allows the robot to learn by itself.

1. However, Karel has later given credit for the word itself to his brother, Josef, in a czech newspaper, *Lidové noviny*, 24 December 1933

Figure 1: *Scene of R.U.R. whith a robot on the left (played by a human), and a woman one the right. In fact, the first ever "robot" is an android because of the human shape.*

This theory is in fact inspired by the biology, where infants in general and human babies in more specifics, come to the world with only a limited set of capabilities but learn during their childhood what is necessary to survive. In this context, the human children shows huge progress in many areas, including cognition, with one particularity : the adults and caretakers interact more actively with their youngsters, in particular to teach them [King, 1991]. Thus, we will explore in chapter 2 several key points in children development related to social learning and cultural transmission, theories that we could then apply to the service robot. First, we will investigate how they could learn a unique tool : the language, an important milestone in child which unlock several possibilities for their learning, in particular to coordinate in collaborative events [Tomasello, 2008]. During these periods or plays, adults could teach them new actions, using imitation, demonstration or instructions using joint attention and the intentional stance that the children is taking. Eventually, the children will then need to keep these events, or any other, in memory if he wants to be able to remember and keep the knowledge he could learn. The autobiographical memory, based on both episodic and semantic memory [Tulving et al., 1988, Conway and Pleydell-Pearce, 2000, Cohen and Conway, 2007], will be explained and we will look for its emerging and developing in children. The information within our autobiographical events could then be retrieved, and using a teleological stance [Gergely et al., 1995] the children could infer about actions, their goals or their constraints.

The chapter 3 will present a study about the adaptation made from language comprehension and production in children development, implemented in our humanoid robot iCub, using grammatical construction [Bencini and Goldberg, 2000] to do the mapping between syntaxic structure of the sentence and meaning in the scene. We will show in chapter 4 a study which explains how this language understanding could be used to co-

ordinate a human and a robot in shared plan cooperative tasks, in particular to mediate precise sub-actions learning through different modalities. Eventually, a model of Autobiographical Memory for the iCub is explained in chapter 5, with an implementation and integration of a classical Artificial Intelligence approach, the Planning Domain Definition Language, to reason about actions and achieve planning, and solve for instance the Tower of Hanoi problem.

Indeed, I will try to cover several sub-domains in the robotics field, in order to provide insight and implementation about needed steps in the developmental approach. I will adress the problem of understanding and producing language by listening to a human, to give next the robot more possibilities to interact with him. The robot will know nothing at first about grammatical constructions, and human input will be only examples of utterance to describe situation, in a natural interaction, without any explicit explanation or teaching. However, the robot will already have developped capabilities in this domain, and thus a speech processing system to extract words from an utterance speech is embedded. He is also capable of understanding situation and could extract relevant information about the world, providing him the references linked to the human language. Thus, I will not go deeply into symbolic grounding problem, linked to meaning and categorization [Harnad, 1990, Rosch, 1973, Gallese and Lakoff, 2005]

Next, I present a system to allows the robot to acquire and learn new motor skills from the human, using the language to coordinate the different layers of the teaching (the shared plan or a precise action), allowing the human to switch through different modalities during the episode. I focus my work on the notion of collaborative tasks, and provide a natural way for a human partner to interact with the robot in order to help him and explain to the machine in real-time if it is needed : what sequence of actions to execute and when. This allows the human to be involved in the development of the robot by interacting with him, which helps the robot for future similar cooperative tasks : the enactive aspect is then more related to the interactions between the human and the robot than to the developmental machine itself. On the other hand, despite the fact that an action learning system is implemented, this part is here constrained by the fact that the robot has to help the human on the fly, and thus only one learning input is provided. Thus action learned in this way are in fact triggering the sequence of motors joint defined by the human (except when he uses already well defined actions using the instruction mode). The action definition problem in particular to adapt the gesture to the actual environment, require more repetition and will not be adressed here.

However, action will be the central component of the last part, in particular related to the extraction of their pre-condition or effects. Using the past experience of the robot and some reasoning capabilities, we implemented to the robot the possibility to learn from previous lived events in order to extract these informations. We are providing here a mechanism to give more flexibility to the robot, and the possibility for him to find a solution by itself when confronted to a problem, even in a not previsouly encountered situation without any help for the human. The whole loop, from the goal to achieve (given by the human using language) to the actual execution of the planned solution has been implemented. But the planning system by itself (using the Program Domain Definition Language) has been taken off-the-shelf : our purpose here is to provide an actual way to recruted already known and well defined Artificial Intelligence solution, and to use them in a real robot with a real-time constraint.

Part I

General Introduction

Chapter 1

From ancient Automata to Service Robots

As we have seen previously, the word by itself is from 1920, but if we stick to these terms, and if we are not too hard on the "complexity" of the tasks, we could in fact travel much more in time than 1920 to study the robot ancestors : the automata, starting from the Antiquity to the Modern era. We will present the most famous of them, trying to identify their strengths but also their weaknesses, and explaining why they are called automata and not robot. We will then introduce the first human-built robots that will show the growing interest in the field as well as the numerous "specialities" emerging from it. Eventually, we will investigate one precise aspect of the robotics that is concerned by this thesis : the service robots, especially with a developmental approach.

1.1 Ancient Automata

If we look at the newest robots, because they are in the state of the art, they are complex machines with a great number of intrinsec properties and different sub-fields covered. Indeed, it is not very simple to try to know what a robot is if we take the last implementation of them. Thus, in order to understand better this specific field, we will travel in time to show the root of it. The journey begins in Greece, during the Antiquity.

1.1.1 Automata from the Antiquity

The oldest known automata was created by the Greek Engineer Ctesibius (c. 270 B.C.) : using pneumatics and hydraulics devices, he produced in particular a water clock as shown in Figure 1.1, with a self-regulated controller and a moving-pointer to mark hours. This is the first known feedback control mechanism [Valavanis et al., 2007].

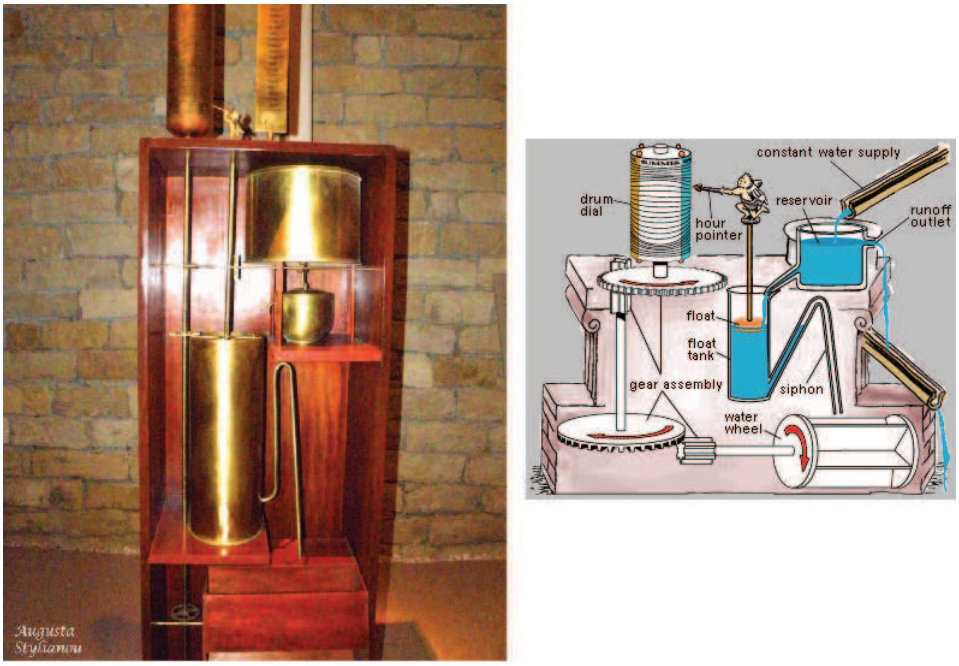


Figure 1.1: *Water Clock with Ctesibius control mechanism (left) and schema (right). The main jar (reservoir) needs to always be full in order for the water to go out at a constant rate. A second tank fill the reservoir to assure that. Eventually, a third container is equipped with a float and a pointer to indicate specific time in an automatic revolving cylinder (because of the gear and wheel, powered by the water through the siphon). (left) Replicas and Reconstruction by Prof. Kostas Kotsanas, from ww.mlhanas.de (right) From edu.ajlc.waterloo.on.ca*

Ctesibus had moreover influenced another great engineer, Hero of Alexandria (who may in fact be one of his student) with a lot of automata, described in *On Automatic Theaters, On Pneumatics and On Mechanics* [Hero, c. A.D. 85]. Working with several kinds of energy (gravity, fire, wind, water, ...) he had made complex machines, included *hypagon* automaton, a moving scene which can bring mobile dolls [Xagoraris, 1991] shown in Figure 1.2. The platform was in particular able to navigate in several kinds of motion (linear, circular, ...) because of a double axis, with parts moved by different weights. By changing the weight, the pedestal could indeed move in a different ways.

With these two examples, we could illustrate what an automata is, using for instance this definition : *"An automaton (Greek, 'self-mover') is a mechanical device which (after releasing a brake) executes a function on its own and in a completely determined way"* [Rosheim, 1994]. The main properties for an automata is then to be able to move. But we could also point two key features:

- The technology behind is part of the definition. Automata uses mechanical theories. That means that, not only the result (moving) is important, but also the way to achieve it.
- It has to be deterministic motions, so it is not programmable. However, this is not in fact a so clear statement : for Hero's theater, one could argue that we can change the weight, and thus modified the produced move. We will investigate further this in a further section.

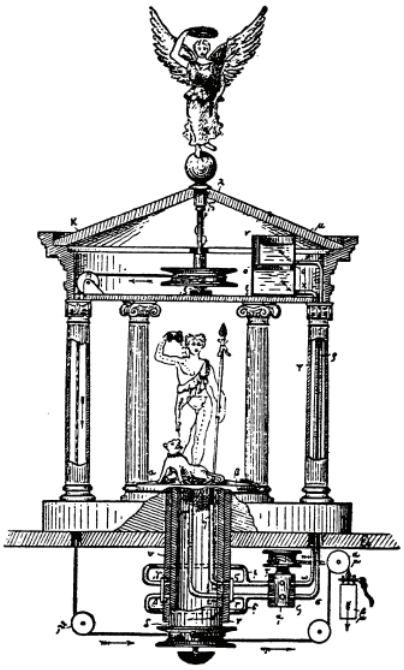


Figure 1.2: *Hero's theater automaton, with Dionysus (near the panther) and Nike (on top). Bacchantes and two altars are near the scence (not showed here). During the play, Dionysus turns to light the fire in the altar and Nike revolves. Bacchantes rotates around the pedestal and also around themselves, with the whole platform moving with several patterns. (From [Xagoraris, 1991]).*

Indeed, some could argue that this was the first programmable machine (at least with predetermined programming, [Struijk, 2011]), but we have to wait for the Medieval Era to have an automata with an "easy" built-in programming feature, and other major advancement in the field, in particular related to the improvement of this capability.

1.1.2 Medieval Automata

We are now in 850 A.D., in the Arabic world, with the Banu Musa brothers (Ahmad, Muhammad and Hasan bin Musa ibn Shakir) and their *Kitab al-Hiyal The Book of Ingenious Devices*, describing 100 inventions [Musa, 1979]. Their work was mostly based of water pressure. Amongst these automata, one is very special, because it is the first programmable machine : the flute player [Farmer, 1931, Koetsier, 2001]. With a schema shown in Figure 1.3, it was based on cylinder with raised pins : by changing them, you could change the melody played [Fowler, 1967], like an ancestor of the punch bands. This is really remarkable because the whole system was design for this purpose : obtaining several kinds of melody by simply changing a piece of the machine. Moreover the pins allow to predict without mathematical calculation the obtained output, contrary to the weights of the Hero's theater.

One other great arabic engineer, inspired in fact by the Banu brothers work, is Al-Jazari (Abu al-'Iz Ibn Isma'il ibn al-Razaz al-Jazari, late 12th century-early 13th century CE). He wrote in 1206 the *Book of Knowledge of Ingenious Mechanical Devices* describing

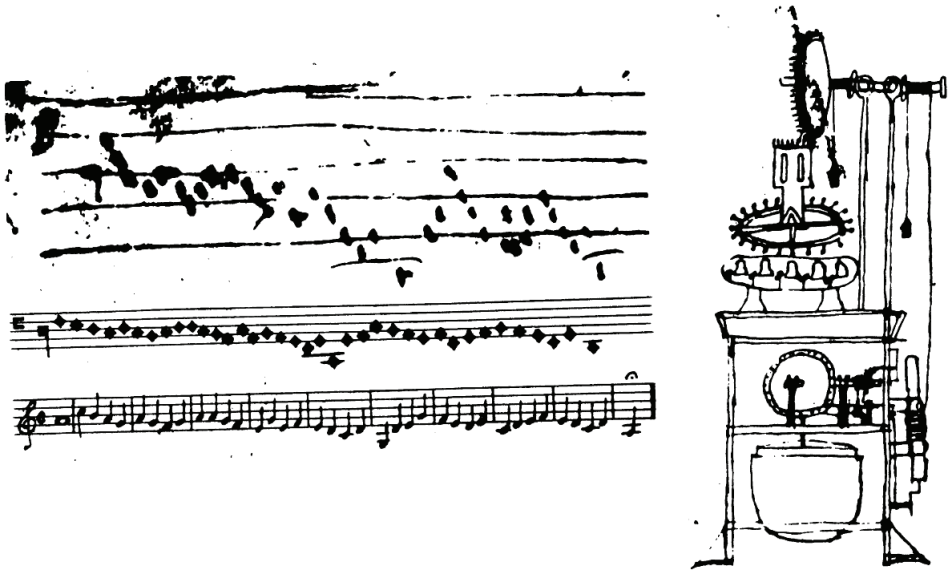


Figure 1.3: *The Flute Player schema of the Musa brothers, reconstructed by [Farmer, 1931] (From [Koetsier, 2001])*

devices using water or air as power system [al Jazari, 1206].

Several of these machines have particular interesting features. In first, the *Arbiter (Hakama) for a Driking Session*, shown in Figure 1.4. It is composed by three separated automata with first a girl, then a castel with five persona (including a dancer) and eventually an upper castle where are a rider and his horse [Nadajaran, 2008]. According to al-Jazari's book, a long time (about 20 minutes) flows until the automata starts to move, after been placed in the middle of the thirsty crowd. Then, the rider rotates with the other persona play music or dance. When the rider stops he points someone in front of him with his lance and the music ceases. The servant girl (the automata) pours the wine to a gobelet, filling it up. A human servant takes the gobelet, gives it to the pointed person who drinks and it is putted back to the servant automata.

The interesting fact is that this cycle is repetead 20 times, with 20 minutes interval : so, of course it is a long wait if we are thirsty and it is not "efficient" in this way, but it highlights two key features :

- the idea of a "cycle" programming, a loop with a final condition (20 drinks served) leading to an exit behavior. At this point, a figure appeared from the castle on the top, saying with his right hand that the service is over, while his left hand indicates two glasses more.
- the longevity and stability of the system. This medieval automata could work for 400 minutes (more than 6 hours) straight, without human intervention.

These two features come up in several other automata from al-Jazari, like the automated boat with music player automata (with 15 performances separated by 30 minutes interval). Indeed the Arabs push the automata to be able to run safely for a long time without any human intervention. This lead them to look for more robust (and so more sophisticated) solution instead of simple one, using feed-back control, close-loop system,

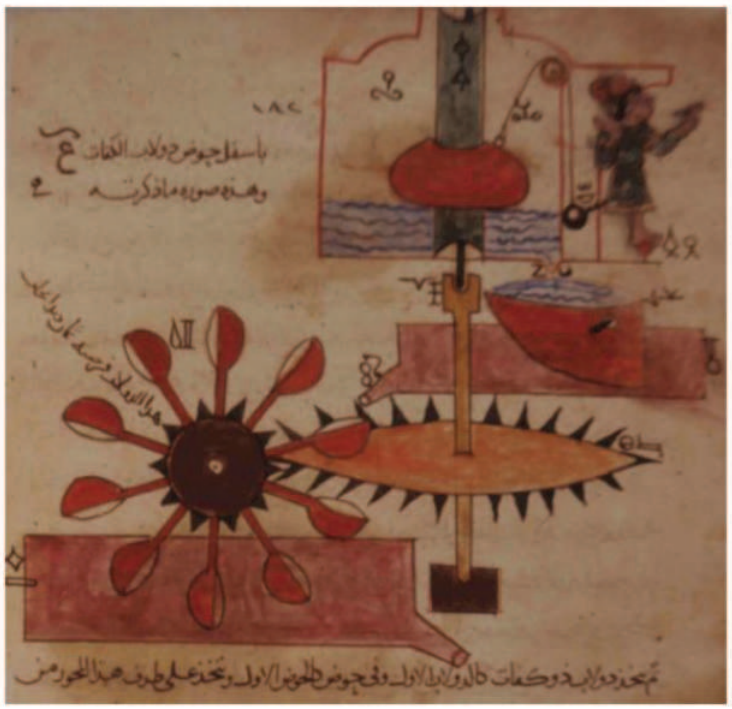


Figure 1.4: *A schema showing a part of the Arbiter for a Drinking Session. (From al-Jazari book).*

... [Hill, 1998].

Another one of al-Jazari's invention has to be described : the two scribes automata for phlebotomy (blood-letting procedure), showed in Figure 1.5. The device was designed in order to give the precise amount of blood taken from the patient. As the blood flows, the scribe in the center rotates with his pen and the board with the measurement is still in front of the patient. But the particularity in this work is that al-Jazari added components and behaviors in order to keep the human distracted during this procedure, showing here an interest in the mental state of the user being bled [Nadajaran, 2008]. This distraction (in addition to the rotation of the scribe) takes the form of an hidden automata, inside the castle, who comes out through one of the 12 automated doors each time a quantity (30 grams) of blood is taken.

Overall and to conclude in this work done by al-Jazari, leading the automata field in the middle-Age in Arabic civilization, we will higlight a last caracteristic feature : the practical applications of the automata. As opposed to the greek automata, mostly focus on entertainment or "special" effect during religious ceremonies, al-Jazari's works have almost always a practical and technical goal to achieve [Rosheim, 1994]. In fact, the book was divides in 6 categories : water clocks, liquid pouring, phlebotomy (blood-letting), fountain and musical automata, water-raising and geometrical tools. As you could see, appart from the foutain and musical parts, every other categories have practical applications. This key element missing in automata from the greek civilization [Rosheim, 1994].

Figure 1.5: *The blood-letting automaton with the 2 scribes. From al-Jazari book (left) and http://www.sciencemuseum.org.uk/ (right)*

1.1.3 Enlightenment era and Modern Automata

One of the main milestone in automata history, and thus a keypoint leading from this field to robotics, is a work done by the well known Leonardo da Vinci during the Renaissance period. Among his countless invention, one is called Leonardo's mechanical knight and has been designed in 1495. An assembly of gears, pulleys, cranks, ... attached to an armor was supposed to be able to move it, controlling several parts of the "body" (neck, shoulder, elbow, hand, wrist for the upper part, hips, legs, knees, ankles for the lower parts). This mechanism was made to produce several human-like actions, including to stand up or sit down, to lift its visor, to wave the arms, ...

I have stated "supposed to be" and "was made to", because in fact Leonardo never build it from the sketches he wrote. But this was done in our time : sketches rediscover by Pedretti in 1957 allowed Rosheim to construct the mechanical knight [Rosheim, 2006], fully operational in 2002 and displayed at different museum as shown in Figure 1.6.

Appart from the achievement by itself, it is also the method and the way of thinking which is very important in Leonardo's studies : the intimately link between human anatomy and mechanism. Indeed, after writing a treatise *elementi macchinali* (mechanicals elements), he moved to anatomical study but by keeping the mechanical concepts, as he stated in corpus at Windsor [Galluzzi, 1987] : *"Arrange so that the book on mechanical elements with its practice preceedes the demonstration of the movements and force of man and of other animals, and by means of these you will be able to prove all of your propositions*". For him, mechanical and organic terms are not to be opposed but instead tightly linked [Garrard, 1987]. He even use mechanical vocabulary to describe or analyze anatomical parts : articulations are shown as revolving axles, analogy between ship's mast and spinal column, muscles as line of force ("power") or levers for motion with limbs.

Figure 1.6: *Leonardo's Lost Robot Knight Exhibit, at the University of Tulsa (2007). From www.leonardoshands.com*

This lead him to quit from the two-dimensional plane anatomical studies, with drawings and schemes, to a three-dimensional support because with this mechanical description, the models could actually be built into functional mechanical devices [Galluzzi, 1987] and could validate his theories. This way of thinking, using biology to build or improve automaton which lead to verify and validate anatomical studies is something that robotics could push and root for.

Indeed, Vaucanson (1709 - 1782), a french engineer, had engaged himself to this path to go beyond. With a background in mechanics, he slides to the study of medecine and to construct a "moving anatomy", with in particular his digesting duck [Bedini, 1964]. Then, he wanted to create an automaton which could move and act as a duck, but also and most importantely, he desired to reproduced also inner process of the living animal : the digestion. Vaucanson's duck was a life-size (Figure 1.7) which could move his wings in an elegant manner (with 400 articulated pieces for each wing) and stretch his neck which was already a well advanced automaton, in particular about imitating a real duck : the moves were designed after studies of natural duck [Riskin, 2003, Landes, 2012].

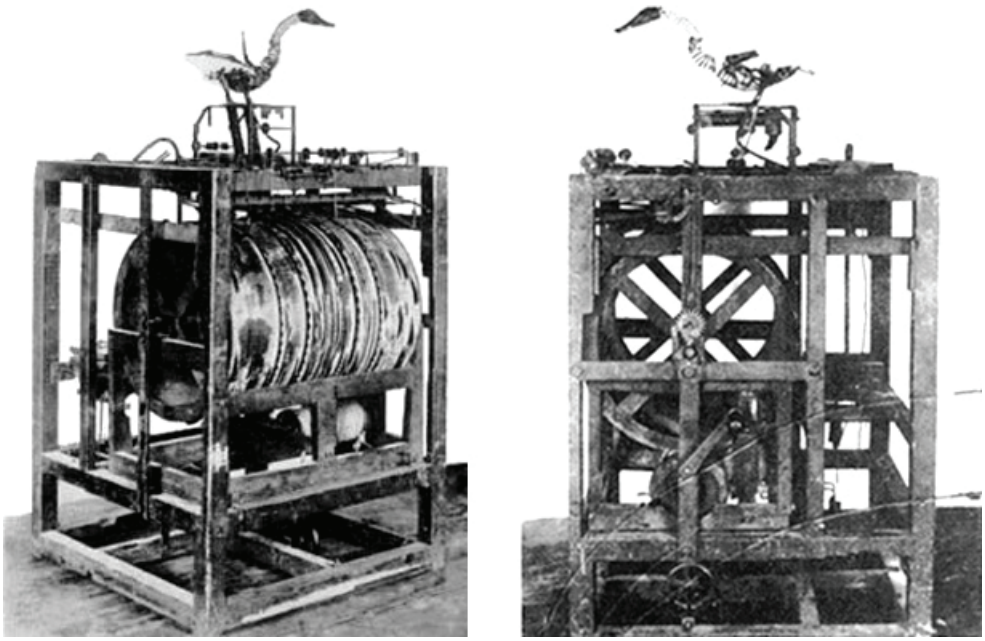


Figure 1.7: *Vaucanson digesting duck, photographs discovered around 1950 in the Musée des Arts et Métiers in Paris, labeled "Pictures of Vaucanson's Duck received from Dresden". (From [Chapuis and Droz, 1958]).*

But it also has the ability to eat grains and, after some time, defecates them. The originality was that the automata body was transparent in the middle to show the process of digestion to the public, using the machine as a tool to teach anatomy [Chapuis and Droz, 1958]. However the biological process was in fact faked, and the duck was keeping the kernels in one container and digested one, stored before the exhibition in another bowl, were excreted after the "eating" part Riskin [2003].

However, if Vaucanson has not fully successed in his entreprise of imitating a process (and not just build a machine), the reputation and fame of this automata has leaded the way for the automata to go from a representation to a simulation (defined by Riskin as "an experimental model from which one can discover properties of the natural subject").

Vaucanson has not stop imitating animal life, but also human life, in particular with his *Flute Player* : life-size (1m65), he could play a dozen of different melodies. Interestingly, the automata really using a flute, it was not a simple "music box" hidden inside the automata (Jacomy in [Spillemaecker and al., 2010]). The tones were produced with the "breath" (with variable air pressure) coming to the instrument through the lips of the android, the fingers moving in order to cover or uncover holes with the inner mechanism showed in Figure 1.8 [Bedini, 1964, Voskuhl, 2013]. So it was not an mechanical imitation of a flute player, but a human-shape automata who actually played flute, like a human do. This leads Vaucanson to discover phenomena ("Discoveries of Things wich could never have been so much as guess'd at", quoting his words) related to the force of the wind required to play : a (surprisingly) great force and an influence of the previous note which change the force for the current note [Seth, 2000]. Voltaire illustrated the fineness of this

work with this quote : *"A rival to Prometheus, [Vaucanson] seemed to steal the heavenly fires in his search to give life".*

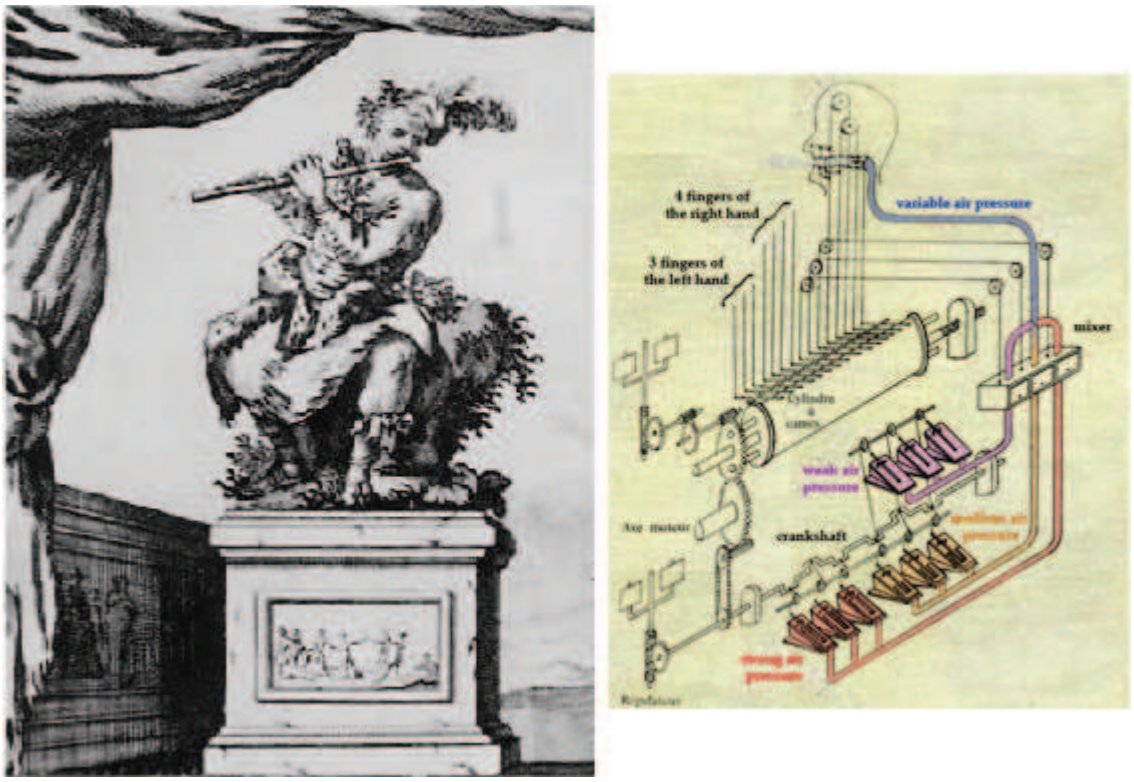


Figure 1.8: *Vaucanson Flute Player, picture and inner mechanism with the variable air pressure which could be expired and both hands moving system, with 4 and 3 fingers (for respectively right and left hand) to play the different tones. (From Dr Barbara J. Becker, lecture 13. Automata. https://eee.uci.edu/clients/bjbecker/NatureandArtifice).*

This goal, to go from machine to "artificial human" will be shared, in particular with a contemporary familiy of clockmakers : the Jaquet-Droz (with Pierre the father and Henri-Louis, the son). They produced three main automata : the musician (female organ player), the writer and the draughtsman, in the 18th century (Figure 1.9), all easily programmable (4 drawings, 5 melodies and more importantly every sentences up to 40 characters).

Figure 1.9: *The Draughtsman, the Musician and the Writer (From left to right), using respectively a lead pencil, an organ and a feather pencil. (From the Neuchâtel Museum of Art and History).*

In these three devices, the automata taken the shape of human (girl for the musician, boy children for the writer and the draughtsman) using their hand to manipulate human tool (like the flute player of Vaucanson), as we do. Thus, this was the most important part, and it was very realistic with the skeleton modeled from real human hands with the help of a surgeon as shown in Figure 1.10 [Perregaux and Perrot, 1916, Riskin, 2003, Moran, 2007].

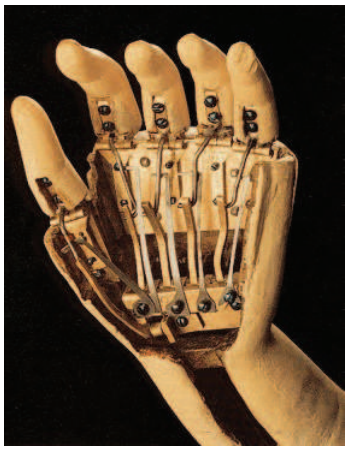


Figure 1.10: *The Musician hand skeleton (From the Neuchâtel Museum of Art and History)*

But why building an entire body, if just the hands are needed in order to achieve the tasks? It is because, like Vaucanson, the "life" of the automata was part of the goal. Indeed, they don't just do what they are built for (draw, play or write) but they do imitating human life : with lift of the chest (breathing), following their hands with their gaze, blowing on the paper to remove dust or bowing at the end, ... ([Landes, 2012], Baldi in[Spillemaecker and al., 2010]).

In conclusion, we have seen that the art of automata has evolved from the Antiquity

era and is not only a self-moving appartus. All the scholar, craftman, engineer, scientists have innovated in this domain and have constructed the foundations of what will become the Robotics. They have built self-moving machines which have practical and real-life applications. Some of them are programmable (at least pre-defined programmation if one want to argue with) and could work for a long period of time without any human interventions. Finally, a part showed an intrinsic relation between the mechanical and physical art of building automata, and the biology, with the ultimous goal to bring machine to life, which act like animal or human and are modeled from them.

They are all requirements to be able to named a system a "robot", but the last one is a property to define a precise field inside robotics which is called bio-inspired. There are still some features missing which will be investigated in the following sections.

1.2 Robots : from Fiction to Reality

1.2.1 General History of Robotics

As seen previously, the term "robot" has been invented in 1920, but we have to wait until 1926 to have one of the first robot, at least the first one which could achieve useful work : Televox. Created by Westinghouse Electric Corporation, it was at first a remotely controlled device to open or close switch and to report the operation to a human, interacting with sound, so not really a robot at this time. But surfing to this new trend, Wensley, the inventor, designed a body to this machine, with arms, legs and head as shown in Figure 1.11. He also added the capacity to lift up a telephone receiver, becoming here a robot which could power up or shutdown machines (ventilator, vacuum cleaner, ...), give information about the states of the switches, ... by interacting with him through the phone [Horáková and Kelemen, 2006, Sharkey and Sharkey, 2009].

Several particularities are to be noted here. First, the "inside" of the robot is not anymore just mechanical : it has electronic composants, removing it from the automata categories to go to the electro-mechanical robot entity. Secondly, the Televox was not a robot at first : without a body to act, he could not "move". It is true that switches could be opened and closed, but not the way human do, it was a simple electronics switch for the Televox. Just adding a body was a great leap from machine to robot, in particular if, with his arm, he could act like a life-being, here, like a human lifting a phone. Last but definitely not least, Televox was designed primary to interact with a human in a "natural" way of speech (tones to be more precise) which could remotely control him. It is an important feature, as stated by Poupyrev [Poupyrev et al., 2007] : *"A significant difference between today's robots and early automata is that the new generation of robots are interactive, designed to understand and respond to people"*

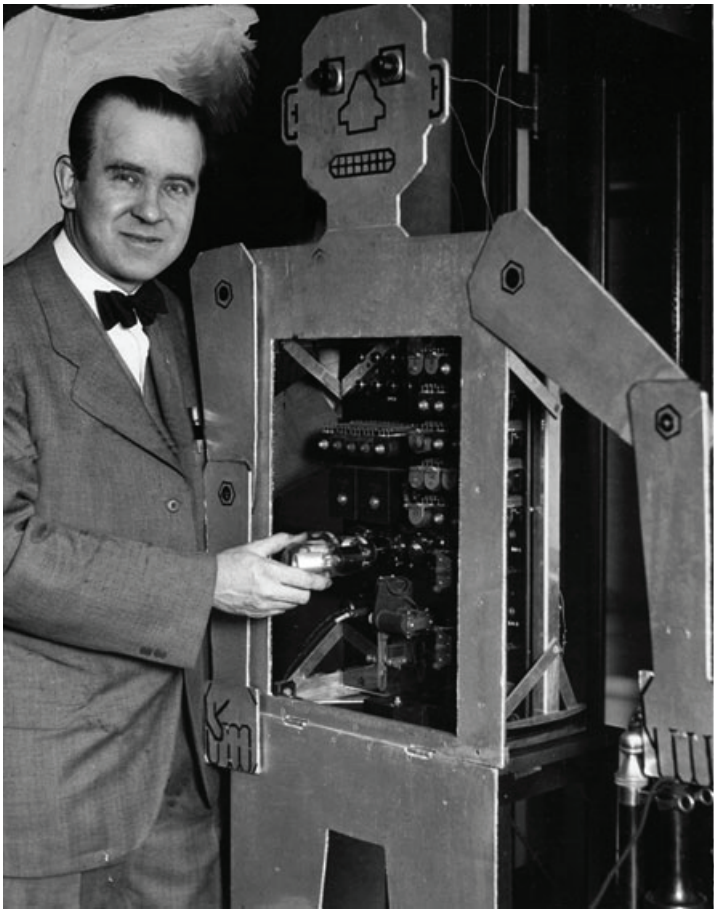


Figure 1.11: *Televox, with his arm ready to lift the phone and his inventor Roy Wensley. (From history-computer.com).*

Westinghouse Electric didn't stop in this field and has improved their Televox to built in 1939 the robot Elektro by the engineer Joseph Barnett. It was a really big (7 feets tall, a little more than 2.10 meters) thin man as shown in Figure 1.12, who could "walk" (sliding with wheel in his foot in fact, but mimiking the walk with the legs), move his arms (in particular to count with his fingers or smoke cigarettes), discriminate colors (between red and green) and has speech recognition (to activate behaviors) and generation (pre-recorded) [Moran, 2007, Ruby et al., 2009].

We "loose" here the direct usefulness of applications from the Televox, but Elektro present a new feature which is very important : it is able to do different operations. And this is a key feature in robotics, to differenciate a robot from an automata. In fact, Joseph F. Engelberger, who founded the first robotics company, has explictely defined this capability : "*An automated machine that does just one thing is not a robot. It is simply automation. A robot should have the capability of handling a range of jobs at a factory*".

Another step has been taken a few years later, in 1948, by William Grey Walter with one of the first electronic autonomous robot : *Machina speculatrix* or more simply known as Grey's turtles [Walter, 1950]. Originally created to uderstand operation in the animal brains, they have been named like this, respecting the biological taxonomy of an animal species to highlight the behavior of exploration of the robot [Fitzpatrick and Metta, 2003].

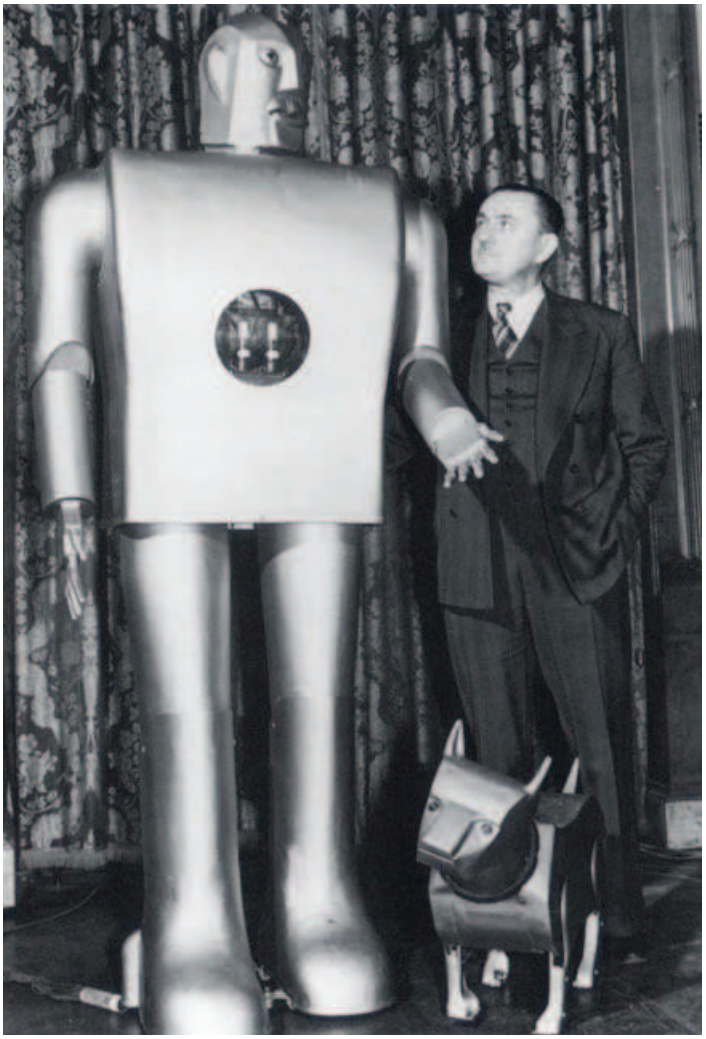


Figure 1.12: *Joseph Barnett and Elektro, with Sparko a robot dog (From history-computer.com)*

It was indeed explicitely described by Walter : "because they illustrate particularly the exploratory, speculative behavior that is so characteristic of most animals".

The *Machina speculatrix* was composed with a battery, radio tubes, two sensors (one for the light, the other for the touch) and two motors (for crawling and steering). All of this was protected by a smooth shell, with the light sensor, a photocell, at the tip of an extension coming out from the body, giving to the robot the appearance of a tortoise, as shown in Figure 1.13

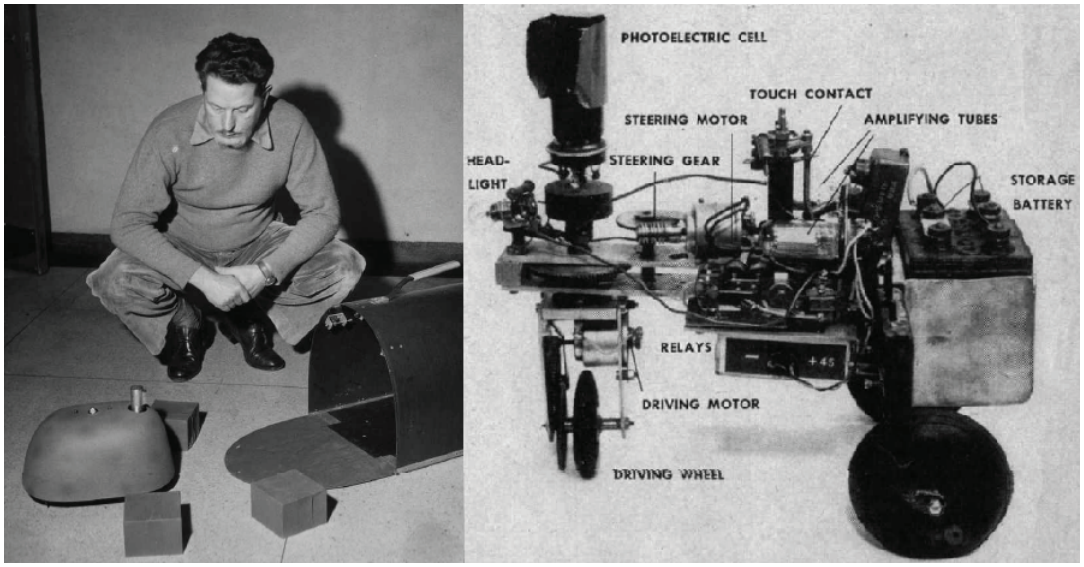


Figure 1.13: *(Left) Walter and a tortoise - (Right) labelled diagram of Elsie, one of the Machina speculatrix, without the shell (From [Holland, 2003])*

Using these sensors (vision, touch and proprioception with the status of the battery), Walter was able to obtained several kinds of behavior according to the data received (as shown in Figure 1.14). They were mainly driven by their photo-receptor (the touch sensor just stops the crawling module when an obstacle is encountered) : when no light is seen, both steering and crawling mechanisms were turned on, resulting in a cycloidal gait. Indeed, the turtle could look in every direction while going forward.
If a light is detected, the exploration module is swith off and the turtle go toward the light, but not too far because if the intensity is too strong, the turtle stops to go in front of her and uses its steering mechanism to go away, avoiding to be "burned". However, in order to recharge his battery, the turtle has to go back to its shelter, where the power supply is close to a light. So when the energy of the turtle is near the end, this maximum light tolerance is removed and the turtle could go in the light and recharge itself autonomously [Walter, 1950, 1951, Holland, 2003].

Walter has with his turtles been a pioneer with his turtles. It was the first self-recharging robot (giving the notion of "survival"), the first attempt in multiple robotics, with experience using two turtles (Elsie and Elmer). And last, but more importantly for us, it was the first biologically inspired robots [Holland, 2003]. Walter robots were built to test biological hypothesis : the complexity of behavior could be obtained not just from the number of "elements" or neurons, but by the way they are linked to each other. Two elements, A and B, could thus produce six dynamic forms : A, B, A+B, A $\longrightarrow$ B, B $\longrightarrow$ A and A $\rightleftharpoons$ B [Walter, 1950]. The turtle are autonomous, showing purposefulness : they are able to look around, detect and avoid obstacle, follow a signal, go back to their shelter, recharge theyr battery, ... Their behavior are not deterministic anymore but depend on their perception and the state of the environement : you could change the position of the light, add obstacles or another turtle. Without reprogramming the *Machina speculatrix*, it is able to manage itself and adapt to these changes.

Walter's turtles are indeed examples of the win-win bi-directional relations between

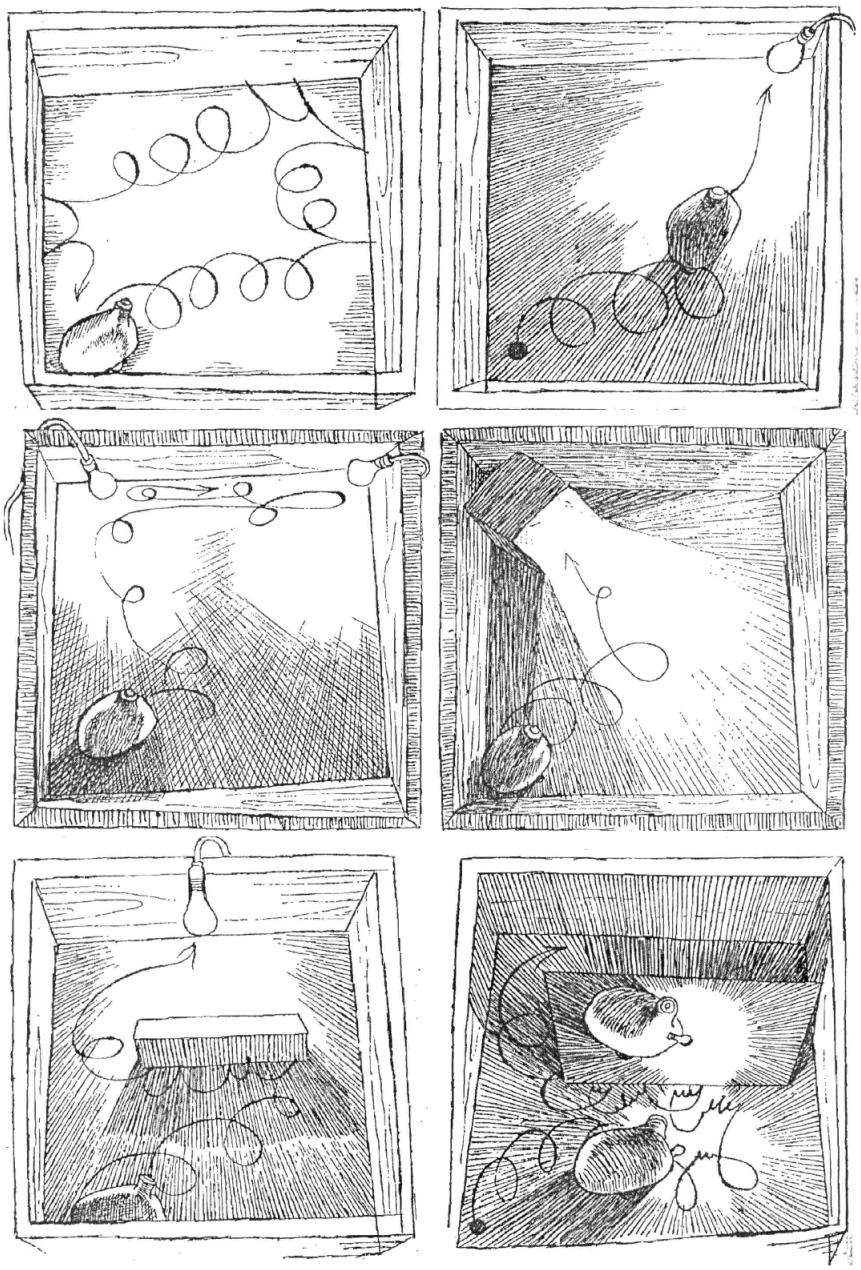


Figure 1.14: *Tortoise behaviors in various conditions. From top to bottom and left to right : tortoise in box, light in box, two lights in box, "kennel" in box, low obstacle and mirror. (From [Walter, 1950]).*

biology and robotics, adapting concept from natural sciences to build good robot and investigate biological hypothesis by using robots [Holland, 2003]. Walter wanted in fact to go beyond this, stating in conclusion of "An Imitation of Life" that *"it would even be feasible to build processes of self-repair and of reproduction into these machines."* leading to an autopoietic (auto for self, poiesis for creation, production) robot, term introduced by Maturana and Varela when trying to define what a living system is [Maturana, 1980].

The *Machina speculatrix* is thus a very good example of another required property

to define a robot. As stated by Heudin (in [Spillemaecker and al., 2010]), about robots "Contrary to these lasts [the automata], they are situated in their environment. In other terms, they can interact with it and adapt their behaviors in consequence". We have here the basis for a robot : a body (to be situated), sensors and actuators to respectively detect changes in the surrounding area and act (interact) according to these changes (adapt).

To conclude this general history about robotics, and to show the huge range in the shape or organization of what a robot is, we will introduce the first industrial robot : Unimate. It was designed in 1954 by Devol who will associate himself with Engelberger to found Unimation, Inc : the first industrial robotic firm was born. Short name for "Universal Automation", Unimate will be put to work inside an assembly line of General Motors in 1961 [Marsh, 2004, Salichs and Balaguer, 2003]. It was in fact a robotic arm used to move hot die casting, poisonous for human worker, and welding them on cars as shown in Figure 1.15 [Henderson, 2006, Engelberger, 1980].

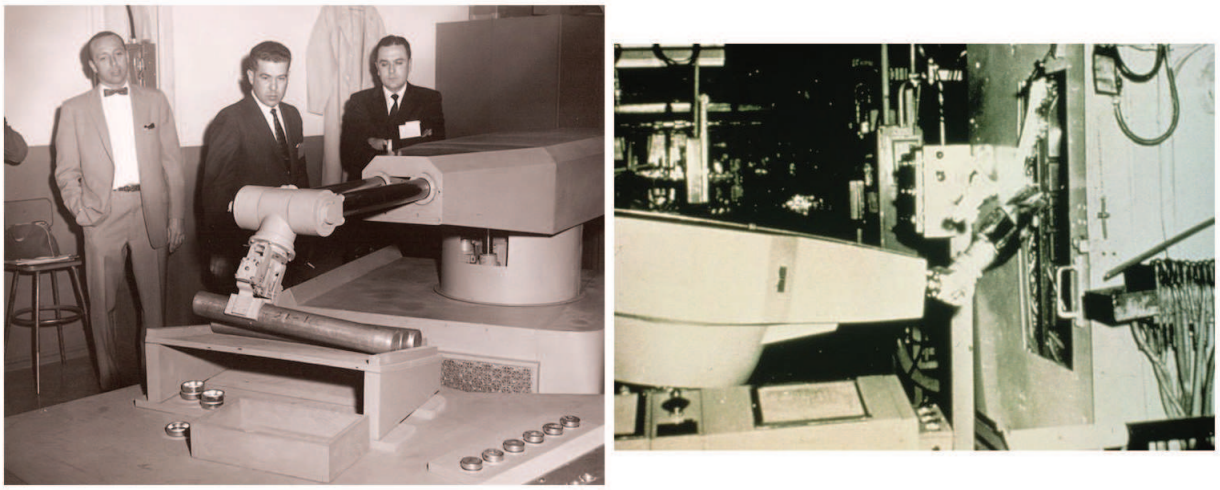


Figure 1.15: *Unimate robot, displayed in a demonstration (Left) and in an assembly line at the General Motor plant in Trenton, New Jersey (Right). (From International Federation or Robotics, http://www.ifr.org/history/).*

In addition to the direct industrial application, one of the particularity of Unimate was its capacity to be flexible. Indeed, he could learn how to perform : using joint coordinates (the angle value for all the degrees of freedom), one could manipulate directly the arm which records the different values taken, to be able to reproduce them after [Singh et al., 2013]. This flexibility has been highlighten by Devol himself in his patent : *"The present invention makes available for the first time a more or less general prupose machine that has universal application to a vast diversity of applications where cyclic control is desired"*

Through this brief history on automation and robotics, and despite the fact we have seen just a few of the hundreds of machines built by the human, we have been able to draw a landscape of the field and more importantly to find caracteristic properties which tend to transform a simple "machine" into a robot. With these elements, we are now able to focus on the definition by itself of a robot.

1.2.2 Robot Definitions

We have already introduced the difficulty to define properly what is a robot. In order to investigate what kinds of properties are relevant and required, we have travelled through history to find the core capabilities or concepts demonstrated by remarkable automata or robots. The list of property shown by a system is in fact the way some entities has chosen to define a robot.

At first, the Robotic Industries Association (RIA) which says that *"A robot is a reprogrammable, multifunctional manipulator designed to move material, parts, tools or specialized devices through variable programmed motions for the performance of a variety of tasks"*

We have already seen these notions through our survey. In particular, the multifunctional properties come from Engelberger way of thinking about robotics. Automata could sometimes move differently, achieve several behaviors but each time, the performance is "global" : the motions are chained and linked between them. Moreover, there is just one purpose at the end : entertainment, service, ... so one function, despite the fact that it could involves different kind of actions. As for the reprogrammability, we have seen that automas could have some "crude" notion of these, so we need to be more precise in the definition itself of this term : the ISO 8373:2012 standard defines it as "designed so that the programmed motions or auxiliary functions can be changed without physical alteration". However, physical alteration is *"alteration of the mechaincal system"* which *"does not include storage media, ROMs, etc."*. Indeed, Hero's theater is not reprogrammable because we need to change the machine (the weights), but the Droz's automata are (but they lack the multifunctional properties).

Other organisations find another solution to define robots : forgetting the universal definition in order to build different class of precise robotics. For instance, the JApan Robot Association (JARA) uses six different ones : manual handling device, fixed sequence robot, variable sequence robot, playback robot, numerical control robot and intelligent robot. It is interesting to note that the first and second classes are not considered as robot by the RIA (because it is respectively directly control by an operator and it is not reprogrammable).

Finally, we will check the ISO 8373:2012 entry. It defines a robot as *"actuated mechanism programmable in two or more axes (4.3) with a degree of autonomy (2.2), moving within its environment, to perform intended tasks"*. We keep the programmable (and reprogrammable) properties, the multifunctional one. We add here the autonomy notion : *"ability to perform intended tasks based on current state and sensing, without human intervention"*. Some automata has touched these (the blood letting device of Al-Jazari for instance) but we could argue that it was only proprioception, the "sensing" was inside the machine : the behavior was related to the current state but not sensing the external world. This autonoma however has been greatly illustrated by Grey's Turtles.

Despite the fact that these definitions are quite robust and general, the ISO norm has also taken the side of different robotics fields definition : robotic device, industrial robot and service robot. The robotic device is not really a robot because it has some but not all of the required properties (in particular the degree of autonomy). Interestingly, the difference between an industrial robot and a service robot is not because of the machine itself, but depending of the goal : the first one is used for industrial automation applications,

the second one for helping humans or equipment (but not industrial ones). That means that a same robot could be in both category (but not at the same time) depending on the context and the current task. It is in fact explicitely stated : *"While articulated robots (3.15.5) used in production lines are industrial robots (2.9), similar articulated robots used for serving food are service robots (2.10)".*

In this thesis, we are focused on the robotics of services. But in this case, what are the differences between them and the industrial ones, if a solely robot could be in both? Whereas the border is thin, and exchange, collaboration and knowledge transfer is possible and occured, the differences is the questions asked and the precise technique primary focused on. The industrial robot tends to be alone, work in a separate environment with safety distance and security from the human workers, whereas the service robot has to be close to the human, to interact with him and possibily to touch and manipulate him. Indeed, we try to build and program robots in order to put them among us, aiming a lot of interactions with them. Instead of having a well defined environment, with mostly inanimate objects, these robot are designed to be in undefined and very diverse areas with animate human beings, manipulating untangible notion as language or emotions and the need to be autonomous.

1.2.3 Service Robot

Service robots are indeed a special kind of robot. They do not have to completely outperformed a human in term of motion (with speed, precision, strength and stamina) but instead, has to offer him assistance in many different tasks. A big part of the complexity in this domain is that the environment in which the robot will be, the precise action he has to do, the people whith whom to interact, etc... are almost completely undefined. Yet, it will surely be put inside a home at some point : but how is this place? How many rooms? Are there some stairs? Is the floor made of wood? Is there a carpet somewhere? Of course, we want him to be able to clean a room, but how? Is there a vacuum cleaner or a broom? Does it need to clean also the windows? Obviously there will be someone in charge of the robot. Is he alone? How old or big is he? Does he speak english?

With these little examples, you could see that trying to model in advance everything and code the proper behaviors and features accordingly could easily lead to a dead-end : we can not predict every difficulties and situations the service robot will encountered because of this open world and multitask purposes [Meeden and Blank, 2006, Asada et al., 2001, Stoytchev, 2009]. Instead of this typical direct programming, we take the approach in this thesis of what is commonly called developmental robotics, or also autonomous mental development methodology or epigenetic robotics [1]. The idea is to implement in a robot models and theories coming from the child's or animal's development science, in order to give the robot the capacity to increase, develop and complexify his cognitive and motor skills through the interaction with its environment [Lungarella et al., 2003, Zlatev and Balkenius, 2001, Prince and Demiris, 2003, Cangelosi et al., 2010, Weng et al., 2001]. In the next chapter, we will then develop these psychological and biological cognitive theories, especially focusing on human child, and retrieve the key elements required to understand

1. There is in fact a small difference between epigenetic and developmental : the first is more interested in motion and morphological development while the last one is more focused on cognitive and social development [Lungarella et al., 2003]

and implement them inside a social robotic platform.

Chapter 2

Toward Adaptive Robotics : the Developmental approach and the Child Development

2.1 Overview of the field

2.1.1 Origin and Definition

This field is relatively new, from the end of the 20th century [Weng et al., 2001, Lungarella et al., 2003] but it takes its roots from 1950 with Alan Turing himself often refered as the father of computer science and artificial intelligence, stating in *Computing Machinery and Intelligence* : "*Instead of trying to produce a programme to simulate the adult mind, why not rather try to produce one which simulates the child's? If this were then subjected to an appropriate course of education one would obtain the adult brain.*" [Turing, 1950]. To be able to obtain this human intelligence in an artificial system, three features are mandatory : the system must have a body (embodiment property), it has to be situated in a physical and social world (situatedness property) and it need a mechanism to increase his knowledge through the interaction with the world (epigenetic developmental process property) [Zlatev and Balkenius, 2001].

The early beginning of developmental robotics comes indeed from theory which study in particular these prerequisites : behavior-based robotics, embodied intelligence and enactive approach [Lungarella et al., 2003]. Behavior-based robotics purpose is to use modular behaviors selected and controlled by sensory information without internal representation system [Brooks, 1986, Braitenberg, 1986]. Embodied intelligence go a little further : the robot could learn these or combined them, because it has a body which could affect the world and it detects these effects and regularities with its sensors, giving him the capacity to verify what it does, to test new motor actions and keep the useful ones. To summarize, everything he learns come from the interaction with the environment : cognition is situated [Beer, 1995, Brooks, 1991, Pfeifer et al., 2001, Wilson, 2002]. Eventually, the enactive approaches rely on the work of Varela and Maturana [Varela et al., 1991] which unifies five concepts [Thompson, 2005, Vernon, 2010] :

1. Autonomy : Living beings are autonomous and act while trying to maintain themself;
2. Embodiement : Cognition is embodied, the cognitive system has to be in the world to be able to interact directly with it;

3. Emergence : The cognitive world built by the system is relative to its interaction with the environment, thus behaviors to act on it emerge accordingly to these interactions;
4. Experience : The history of all the interactions has thus a key role for the mind;
5. Sense-Making : The system keeps a coherent pattern according to sensorimotor information, creating meanings between them, detecting regularities in order to discover the laws of the environment.

It has to be noted that the sense-making property of the system emerges from experience. But these interactions are leaded by the autonomous features of the system : it has to maintain himself. Thus, the knowledge built is dependent of both the system himself (with the body constraints) and the explored world and the purpose from this is to incrase the possible space of actions of the robot.

So in conclusion, enactivism adds to the embodied cognition a self-organization feature with two mechanisms : co-determination and co-development [Vernon, 2010, Metta et al., 2008]. Co-determination is where the system builds the world according to the perceived consequences of his actions on the environment. Co-development means that the knowledge learned emerges from all the possible actions the system has done through its experience, in particular with the invariances or regularities which occurs.

Thus, a key feature for this approach is the interaction between the robot and the world. But this world in not only composed of inanimate objects, there are also a special kind of entities that the robot will encountered in his surroundings : animate agents such as humans. Indeed, humans are not only physically situated in the world, they are among other agents, involved also in a social and cultural world [Brooks and Stein, 1994, Edmonds, 1999]. Alan Turing's child is not alone with his toys or dolls. He has parents, siblings, caretaker with him, allowing him to use and learn already known knowledge from them : the cognitive development is greatly impacted by social factors and interaction with other humans [Vygotskij, 1934, Whiten, 2000, Meltzoff and Prinz, 2002, Tomasello, 2009]. It is one particularity of the human species : the caregivers interact actively with the youngster to help them to gather and organize the information needed as opposed to the other primates species, where adults intervene not often [King, 1991]. It is defined as the social or cultural hypothesis which explained the difference in cognitive skills between the human and others species (included the nearest primate relatives) by early ontongenic development of specific social skills, as shown with an extensive comparative studies between human, chimpanzees and orangutan in both physical domain (not difference) and social domain (human children advantages) in the Figure 2.1 [Herrmann et al., 2007]. It has to be noted that this hypothesis is an extension of the more general social intelligence hypothesis, which is not only applied to human species but also to other species with social group like some apes [Whiten et al., 1999, Whiten and Van Schaik, 2007, Van Schaik et al., 2003]. The difference lies on the way the cultural interactions are made, with three particularities for humans [Herrmann et al., 2007]:

1. Learning a language of their cultural group through social interactions ;
2. Acquiring subsistence capacities from experts ;
3. Developping skills through schooling (written language and mathematical symbols).

The social contact is thus very important for the children, a lack or impairement in the social or communicative skills could lead to development disorders, encountered in particular in autistic children [Baron-Cohen, 1997, Scassellati, 2001]

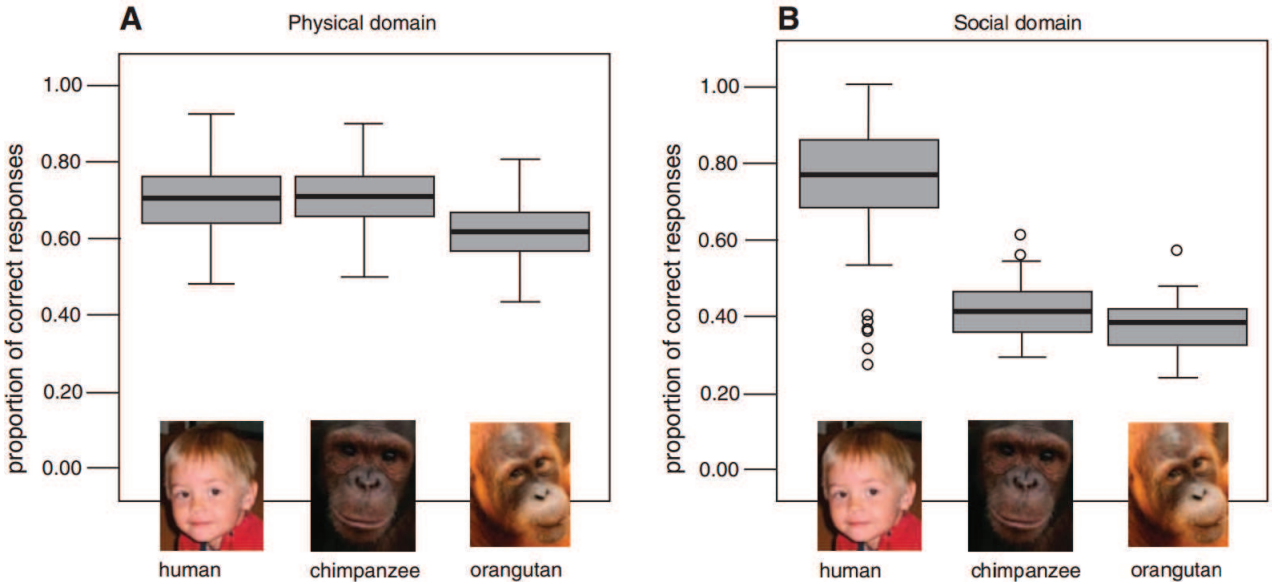


Figure 2.1: *Overall performance in Physical domain (A) and Social domain (B) of the Primate Cognition Test Battery for human children, Chimpanzees and Orangutans. The test is built with 16 tasks grouped by cognitive scales : space, quantities and causality for Physical domain, Social learning, Communication and Theory of Mind for the Social domain. Children are 2.5 years old. (From [Herrmann et al., 2007]).*

That is why we will focus in this part to this special kind of learning, where a teacher is available and interact with the immature being, a children or a new robot, to help him go through his development.

2.1.2 Social Learning and Cultural Transmission

This social situadness of children - which has recently led to taking this aspect into account in cognitive science and artificial intelligence theories, in particular for robot [Brooks and Stein, 1994, Dautenhahn, 1995] - take in fact it root in the first part of the 20th century with work by Vygotsky and Piaget with the cognitivism learning theory. The learner will build knowledge with a sequential development, acquiring abilities and information from the environment but also from people. More precisely, adults can i) help the children to organize or combine skills, or reduce difficult of a task by guiding the attention of the children, setting up easier intermediate steps, ... (called scaffolding by [Wood et al., 1976, Lungarella et al., 2003]) or ii) by direct instruction [Kruger and Tomasello, 1996].

This cognitive development will then allow the children to build, integrate and use mental functions. Vygotsky introduces in this mental development the notion of Zone of Proximal Development (ZPD), defined as the distance between what the child could do by itself, unaided, and what he is able to do with adult guidance, cooperation with other [Vygotskij, 1934, Lindblom and Ziemke, 2003], as shown in the Figure 2.2. He argues that an infant could only imitate, learn or understand things or concept inside his ZPD, with too complex notion impossible to catch despite a potentially great number of repetition.

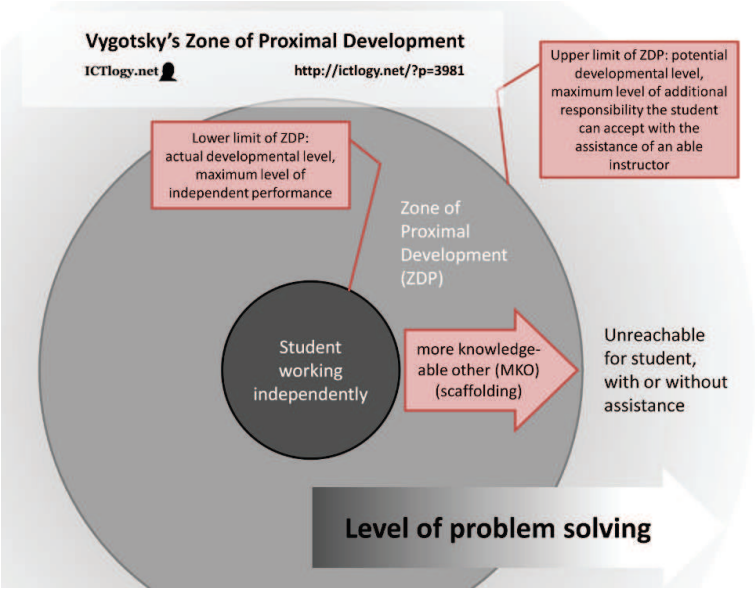


Figure 2.2: *Zone of Proximal Development schema, between the independence and assisted performance area according to the difficulty of the task. (From Peña-López, in http://ictlogy.net/)*

Vygotsky make also a disctinction between the mental functions : the elementary and the higher ones. According to him, the first functions are "innate" and could be found in other species apart from human, for instance perception, attention or simple memory. The higher ones are specific to human and emerge in a non direct stimulus-response process but with an undirect process because of what he called an intermediate link (physcological tool) between the stimulus and the response.

Indeed, neonates already demonstrate some skills, in particular in the social domain, such as protoconversation or mimic movements [Trevarthen, 1979, Meltzoff and Moore, 1977]. Moreover, between 9 and 12 months, infant understand that others (parents, siblings, caretakers, ...) are intentional beings with goals, allowing joint attention and engagement and thus triadic interaction between the child, the caretaker and the object of focus [Tomasello, 2009]. These early abilities will be the base for the direct instructions and the scaffolding mediated by the teachers, allowing to shift the ZPD of the child over time, according to his development as shown in the Figure 2.3 [Leong, 1998].

Thus, one skill will be particularly important for the child during these interaction : language understanding and production. Language will be used extensively by the teacher in direct instruction, but also when he needs to help the infant in his purpose, for instance by leading the attention to a key object by saying the name of it. It will also allows both agents to coordinate in order to achieve more complex actions and shared plans, where each will be responsible of precise action in a definite order to complete something which could not be done alone [Ashley and Tomasello, 1998, Tomasello et al., 2005]. It could be also to make the task easier for the children where the teacher takes care of the more difficult one, keeping the learner's tasks within the zone of proximal development of Vygotsky.

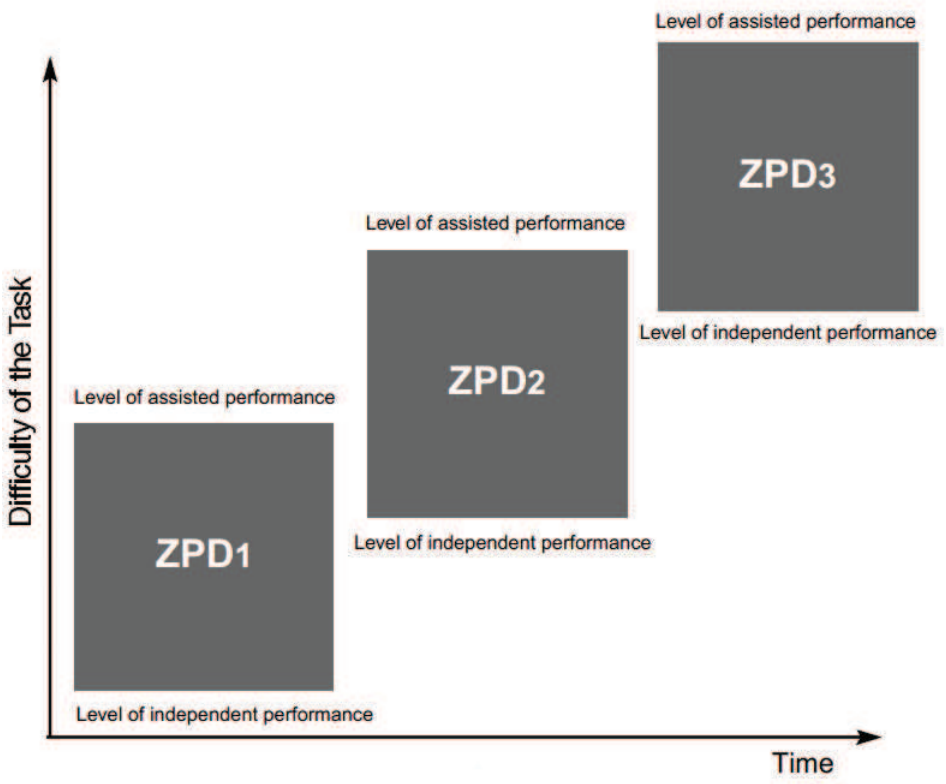


Figure 2.3: *Shifting of the ZPD through the acquisition and development of skills. (From [Leong, 1998]).*

2.2 Child Development : Learning a language

Language comprehension and production is thus an important milestone in child development, allowing him to use in the most effectiveness the cultural transmission, in particular for coordination activity [Tomasello, 2008]. However, the infant acquires before that some other physical or social skills which will in fact be useful for learning his native language.

The most dominant approach in modern linguistic during the second half of the 20^{th} century was the generative approach of Chomsky [Kaplan et al., 2008]. This consist of obtaining a finite set of formal statements or rules which allows by a deductive process to produce only the all possible correct grammatical sentences for a language [Chomsky, 1957, 1959, Halle, 1962]. However, the language acquisition does not not consist of building this grammar from scratch, but instead, parametrizing a Universal Grammar (innate and genetically inherit among humans) to match the native (or any other) language of the children [Chomsky, 1965]. The Universal Grammar was introduced to solve the Poverty of Stimulus (PoS) argument which says that the children is not exposed to enough consistant and complex inputs to allow him to infere rules and to make grammatical generalization [Chomsky, 1965, Stich, 1978]. Among others, one aspect of this Poverty of Stimulus is the fact that children listen to only correct grammar sentences told by adults when they begin to acquire their native language and thus have a lack of negative evidence. And even after some point, when they begin to have a basis and try to produce sentences, including

incorrect ones, the corrective feedback provided by the caretakers tend to be ignored by the child (McNeil1966). And if the child gives attention to caretaker feedbacks, these ones are noisy without a strong contingency between the feedback pattern and the syntactic correctness [Brown and Hanlon, 1970]. And in fact, these signals do not discriminate between grammatical and ungrammatical constructions [Marcus, 1993].

Indeed, Gold showed that it is impossible to learn from positive evidence solely a hierarchical structure (as produced by grammar of human languages) with infinite recursion [Gold, 1967]. Nervetheless, this hypothesis has been challenged more recently with arguments for a "Richness of the Stimulus" [Tomasello, 2000, Sampson, 2002, MacWhinney, 2004].

In particular, the claim that positive evidence only in not enough as input for language acquisition as been challenged by MacWhinney [MacWhinney, 2004] : it is not having negative evidence which is curcial to the child but having enough quantity of good quality positive evidence. Moreover, the poverty of the stimulus itself could be in jeopardy : the child is not just exposed passively to language in order to acquire language, but he will be directed in respect to important elements in the oral language or in the environment to be able to manage the mapping between sentence and meaning, in particular (but not only) with speech modulation from adults and joint attention [Dominey and Dodane, 2004]. This allow a more developmental perspective of the language acquisition skill in children than the classical Chomsky's view, with cues being given to help the infant to manage the relevant aspect of both the speech part or the environement part, to achieve the correct mapping between sentence and meaning.

2.2.1 The Richness of Stimulus : the Prosodic Bootstrapping Hypothesis

This hypothesis relies on the principle that, in fact, the input for language is not poor but instead contains clues about syntactic grouping that reduce the requirement of the needed grammar [Morgan and Demuth, 1996]. Now, the prosodic pattern of the spoken signal is carrying a great amount of information about precisely the syntactic structure of the language : the children is then helped when he has to initially learn and extract this structure [Morgan and Demuth, 1996]. Instead of having a pre-wired universal grammar at birth, the infant just needs to acquire a small set of concrete nouns (without any link to grammatical knowledge) to built a basis which allows him to pair off actions (meanings) with syntax (forms) from the syntactic categories [Gillette et al., 1999, Pinker, 2010]. Indeed, the children's early speech is most exclusively composed with open class word (OCW or semantic word, carrying the content), and the close class word (CCW, or grammatical word, for instance "to", "with", ...) appeared in a correct and systematic way later [Morgan and Demuth, 1996].

Moreover the adult's prosody in not the same if they talk to others adults (ADS, Adult-Directed Speech) or to a child which is called CDS for Child-Directed Speech. In CDS, the segments of the sentence are deformed by exaggerating the prosodic structure resulting in a 'sing-song' language form [Fernald, 1989]. The information carried by these cues are then emphasized and easier for the child's perceptual skills to extract and segment. In fact, 4-months infants show preferences for this kind of 'motherise' speech compared to ADS samples [Fernald, 1985].

This cues encountered in CDS can take several forms, like longer pauses between grammatical subsections [Broen, 1972] or high-pitch tone for new words localized at the end of the sentence [Fernald and Mazzie, 1991]. Indeed, children as young as 6 or 7 months could segment words from a fluent speech principally using statistical information but their strategy is adapting and they mainly used stress syllable cues from 9 months in order to achieve this tasks [Thiessen and Saffran, 2003]. Another possible cue, related to this last one is the 'F0', the fundamental frequency, whose contours are exaggerated [Fisher and Tokura, 1996]. Yet there is a correlation between the F0 and the word classification : a F0 peak is more linked to open class word and an absence of an F0 peak points more toward a closed class word [Dominey and Dodane, 2004].

Indeed, prosody is important when linked to children language acquisition. By modifying the way we talk to infants, we help them to segment or categorize signals in particular by leading the attention of the child to precise and relevant part of the speech signal. It can then be treated more easily in order to extract useful information, for instance being able to classify the word as a CCW or a OCW.

These knowledge about the classification could then allow to use the syntaxic form of the utterance to extract information about, and possibly unknown, OCW. The grammatical construction theory takes this approach and is defined as a mapping between the sentence structure and the event meaning structure, which occured during the speech [Bencini and Goldberg, 2000]. Thus depending on the learnt sentence type, the child is able to know that the position of the word in the sentence correspond to a precise role. For instance, if we encountered a simple active transitive sentence (e.g. 'The octdor dartis the dakel'), the first noun ('octdor') is the agent who do the action ('dartis'), and the second ('dakel') is the object who suffered it [Naigles, 1990, Golinkoff et al., 1996, Dominey and Dodane, 2004]. One hypothesis is that these grammatical constructions, which help to solve uncertainty about new words, can be in particular identified and categorized with the unique constellations of CCW or morphemes, or other cues in the Competition Model of Bates and MacWhinney [Bates et al., 1991].

We have seen here some clues which can be used by the child in order to manage and extract useful information from speech input signal from adults, and use them to acquire knowledge about the sentence. We will now look down to the other parts of the mapping occuring during language acquisition : the meaning.

2.2.2 The Richness of Stimulus : the Joint Attention

The other aspect of the Poverty of Stimulus, concerned about the 'meaning' extraction, is that the visual scene perceived by the infant when he hears a sentence is noisy : what precise aspect or part of this scene is relevant to the utterance told by others? Even if I can manage this, my point of view is not the one of the adult : the perspective is not the same [Quine, 1960]. During this early language acquisition period and related to this problematic, research shows an important role of what is called joint attention [Tomasello and Farrar, 1986, Hood et al., 1998, Morales et al., 2000, Mundy et al., 2007]. It is defined as the competence of the infant to coordinate her focus with another social partner about an event or an object. Sharing an episode of joint attention between a young children and an infant tend to improve the language acquisition process related to the word-object mapping problem, because it helps the infant to identify the relevant

referent of the utterance speech, in the visual scene [Morales et al., 2000, Dominey and Dodane, 2004]. Then it allows the young children to go from dyadic (between him and an adult) to triadic interactions (him, the adult and an object) around the first year of life as shown in Figure 2.4 [Tomasello, 2009]

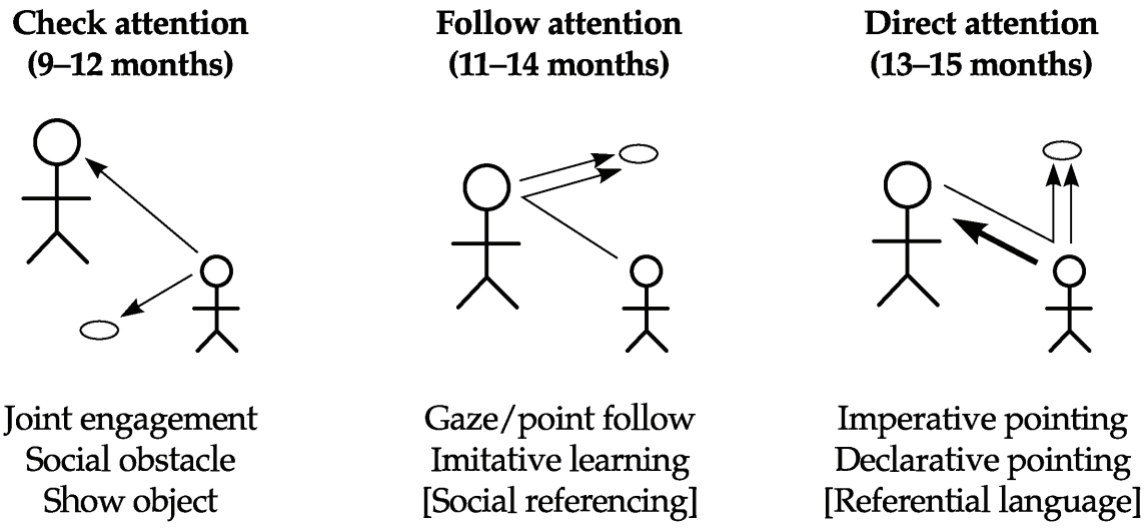


Figure 2.4: *Main joint attentional interaction types and respective age of emergence (roughly 80% of subject inside the range). (From [Tomasello, 2009]).*

However, a joint attention episode could be happening in two ways : the adult could actively direct the attention of the children to something specific he will talk about (by looking or pointing at the focus object for instance) or the caretaker could follow the natural attention of the children and adapt to it, commenting then what the infant is currently gazing at or playing with. Interestingly, despite the fact that the overall RJA in early age (under 18 months) is positively related to the length of the lexical vocabulary acquired by the children [Morales et al., 2000, Brooks and Meltzoff, 2005], the children with parents who followed their babies focus of attention to initiate joint interaction have bigger vocabularies than children of parents who actively redirects the attention of them for this kind of episodes [Tomasello and Farrar, 1986].

However, despite the fact that a causality flows from joint attention to language, it has to be precised that this is not only a one-way causality. Instead, the influence works more in a "transactive" way : joint attention helps the language acquisition of a prelinguistic child which is used in return to optimize and maintain theses joint attentional episode, resulting in more effective language interactions [Tomasello and Farrar, 1986]. It is this precise support and coordination role during interactions, including learning through several modalities, in order to achieve a shared plan, which is based on these precise joint attentional episodes.

2.3 Child Development : Learning through multi-modality, coordinated with language

The step where the children acquire enough lexical vocabulary and grammar skill is a crucial step for the development, in particular because the main function of the language relies on the coordination during cooperative activity with others, which is based on joint attentional skills [Brinck and Gärdenfors, 2003, Tomasello et al., 2005, Tomasello, 2008]. This is crucial because social and cognitive developments is mainly built from cooperative plays with caretakers or other children [Piaget, 1932, Hartup, 1989]. The true and wanted cooperative behaviors emerge from 18 months after the infant achieved a pre-requisite step which is to learn the notion of intentionality, that not only he has but other persons as well [Tomasello, 2009, Brownell et al., 2006]. Then the children can see others as intentional agents, and simulate their goal and desires than true imitation or cooperate shared plan is possible [Tomasello et al., 2005].

2.3.1 Joint attention and self as intentional agents, like others

As we have seen previously in figure 2.4, joint attentional episodes emerge from nine to twelve months and will become more complex and consistent with time, allowing the infant to use not only dyadic interactions (manipulating objects or expressing emotions toward others) but also triadic behaviors with a share attention between him and others toward an object [Tomasello, 2009]. During the same period, another crucial cognitive skill appears : 12-month-old infant can attribute a goal pursued by agent and think about his actions in relation to that purpose, at least related to spatial behavior [Gergely et al., 1995]. Tomasello hypothesis is thus that it is when the infant begins to understand that others are intentional agents like him that the children engages in joint attentional episodes [Tomasello et al., 2005].

Indeed, the equivalences between me and the other, who is 'like me' is the foundation of the social cognition [Meltzoff, 2007a,b]. The goal-oriented actions system involves intentions and emerges from an early sense of self, what Neisser called the "ecological self" [Neisser, 1988]. It is the step where the infant knows that he is a differentiated, situated and agentive entity. This stage could be achieve by exploring and interacting repeatedly with objects to observe the consequences. It comes from early dyadic interactions when 18-week-old children could produce rudimentary reaching behaviors toward a translating object with anticipation [von Hofsten and Fazel-Zandy, 1984] or when 6-week-old infant imitates tongue protrusion from a caretaker [Meltzoff and Keith Moore, 1994].

Thus, the joint attentional episodes and the "like me" intentional stance could be combined and develop each other in a transactive way around the 9th month of life. Children begin to see their behaviors as goal-oriented and the new triadic interactions allow them to project these intentions onto other agents as summarize in Figure 2.5. In return, shared attention could be mediated and facilitated with this new perspective in particular by separating the goal and the intermediate steps or behavioral means to achieve it [Tomasello, 2009]. For instance, children directs the adult attention with visual cues (by pointing) more often when others can see the object, compared to when he cannot.

This attribution of intentions to another person combined to the joint attention development at 9 month will then open new possibilities for the cultural learning of the infant,

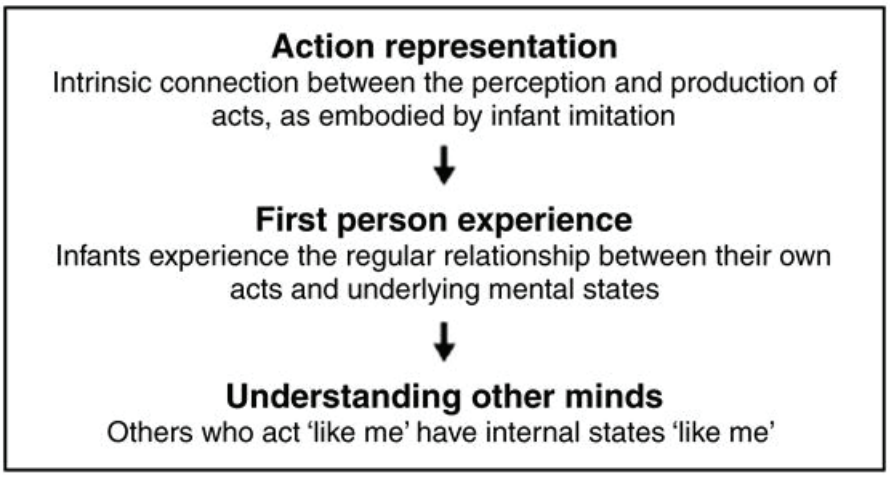


Figure 2.5: *Development of the "like me" framework (from [Meltzoff, 2007a])*

which could be one of three types, the first one will be the imitative learning. We have seen that the children is capable of some kind of imitation before (especially facial one like the tongue protrusion) but it is rather a mimicry : the infant reproduces the behavior he just saw, without any regard for the intention and current goal of the observed agent. For a true imitation, the learner should take the perspective of the user, in particular understand his goal, and then reproduce the majority of the observed behavior . The second one is the instructed learning when the children remember the instructions and demonstrations of the teacher and in which context (i.e. what is the purpose of the behavior) it could be used, thus a certain theory of mind has to be developed to take mental perspective. It could take several forms, like providing a set of instructions using spoken words. The third one is called collaborative learning when both (or more) participants shared a common goal and plan which could not been achieved alone and where cooperation for the different behaviors to do are necessary, with the possibility of role reversal in an episode of shared plan [Tomasello et al., 1993, 2005].

2.3.2 Imitative learning : the true imitation

We first have to define precisely the true imitation, or imitative learning, compared to other imitation-like processes, in particular the emulation. In true imitation, you are learning by copying the actions of others, whereas in emulation you are looking to the results of the action, the property of the tool used (if any) or the objects involved [Tomasello, 1990, Whiten, 2000, Whiten et al., 2009]. Indeed, chimpanzees are better at emulation learning, for instance when the mother rolls a wood log in order to find and eat insects behind, the youngster will learn that removing the lot allows to have access to food, but the "rolling" move will not be exactly copied, because it is already something he can do or learn by itself with trial and error [Tomasello, 2009]. In comparison, human children tend to adopt more often a imitative learning posture as shown by Nagell [Nagell et al., 1993]. In this experiment, chimpanzees and two-year-old children, mixed and separate in two groups, were presented an experimenter with a rake-like tool for an out of range reaching tasks, who used it in a different ways between the groups, one of the method being more efficient than others. Results show that children are trying to copy the method used (imi-

tative learning), whereas the chimpanzees use different kind of strategy to manipulate the rake independently of the method seen (emulation learning). We could see here that there is no "more intelligent" or "better" learning, they are just focusing on a different main property (behaviors for imitation, goals for emulation) and their efficiency depend on the task to achieve or teacher skills. Here for instance, chimpanzee using their own behavior to use the rake was better than children trying to copy the less efficient method from the experimenter. However, what we could state is that imitative learning is more social and allows to learn new behaviors of ways to act and so is a crucial skill for culture transmission.

Indeed, inventive and more efficient ways to act and use tool could be found by an individual, but it could be transmitted to the peers and the following generation, what Tomasello called the cumulative cultural evolution (or ratchett effect, shown in figure 2.6), only if others try to copy the new method, and not only continue to use the tool in the way they are used to do.

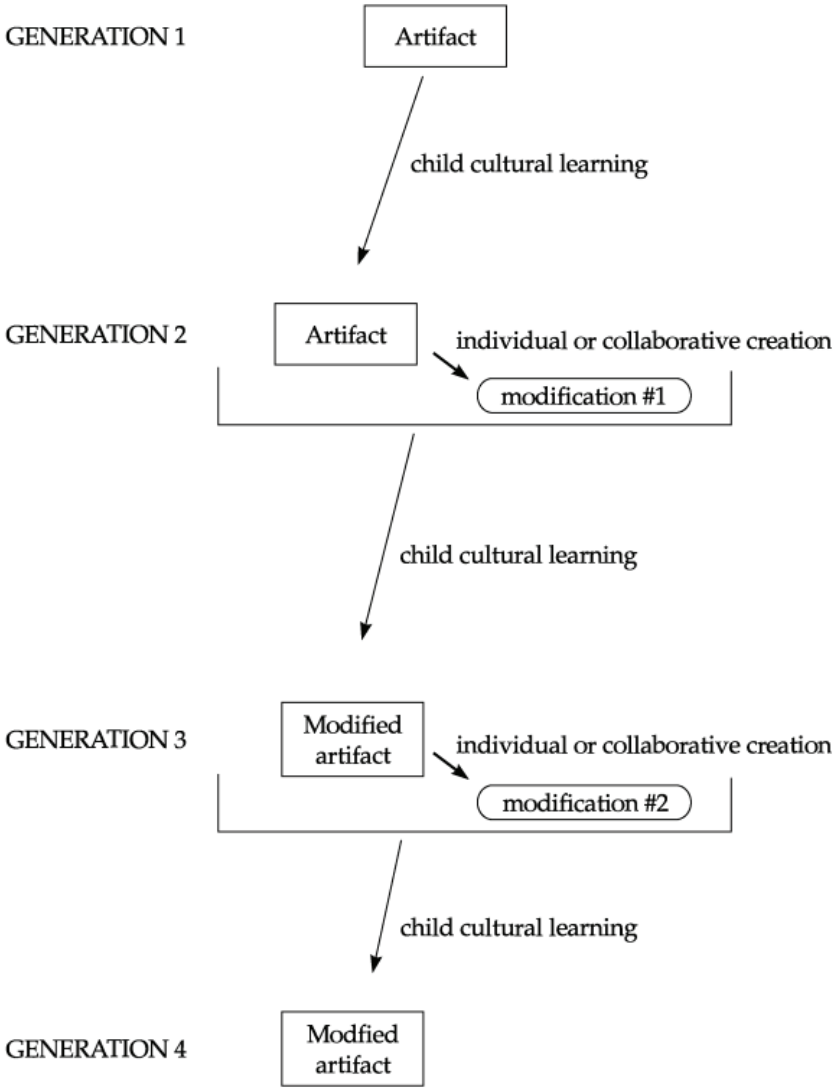


Figure 2.6: *Simplified schema of the ratchett-effect, showing an artifact with cumulative modification. (From [Tomasello, 2009]).*

However, if children tend to prefer true imitation, they are also capable of producing emulation, and in fact they choose between these modalities according to what they understand of the behavior and the goal observed and what aspect is the most important for the teacher. Meltzoff then Carpenter found that 14-month-old children prefer to imitate the unusual and awkward behavior of switching on a special light by touching the interrupter with the head than just simply using their hands ([Meltzoff, 1988, Carpenter et al., 1998]). However, Gergely reproduces this experiment with a new condition : the experiment could have either his hands free or wrapped by a blanket. In fact, 14-month children mainly imitate the weird actions where the adult's hands were available whereas they principally used their hands to switch on the light (emulate) when the demonstrator had his hands occupied as shown in Figure 2.7. The 14-month children can then select the learning mechanism in a inferential process, involving in particular the rationality and constraint of the current situation ([Gergely et al., 2002]).

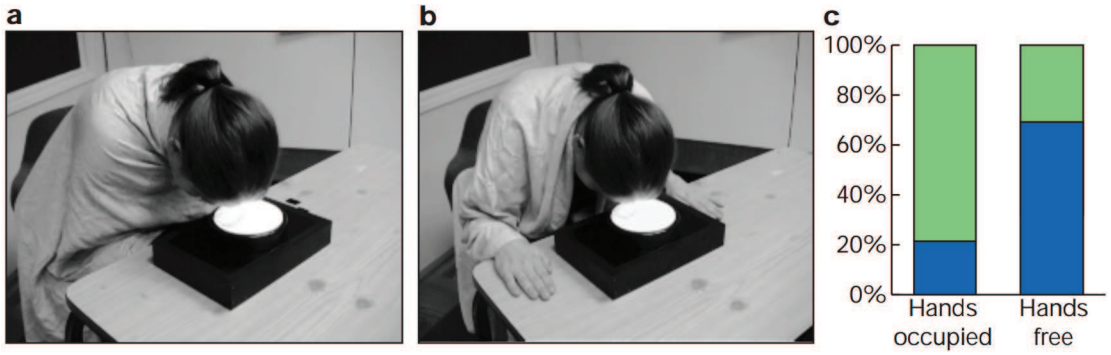


Figure 2.7: *Original Caption : Comparison of the methods used by 14-month-old infants to switch on a light-box 1 week after watching how an adult executed the same task under two different conditions.* ***a, b,*** *Adult switching on the light by touching the lamp with her forehead in the hands-occupied condition (**a**, n = 14) or the hands-free condition (**b**, n=13).* ***c,*** *Methods used by infants to switch on the light-box after watching the head action used by the demonstrator under these two conditions (left bar, adult had hands occupied; right bar, adult had hands free), recorded over a 20-s period. Blue, head action was re-enacted; green, only manual touch was used. (From [Gergely et al., 2002]).*

2.3.3 Instructed Learning : language coordination

Another effect that could have an impact on the selection of imitation or emulation is the behavior of the adult, especially related to socially interactions. Indeed, the imitative learning could be favored, and facilitated, when the teacher uses ostensive referential speech acts like saying "Look at my hand", accompanied by a coherent gaze, smile. Gergely and colleagues have found this result by reproducing their hand-free versus hand-occupied task from 2002 and observing a significant increase in head movements in the first condition, when children receive these pedagogical cues compared to when the teacher is not socially active [Gergely and Csibra, 2005]. Other results confirmed this tendency to copying actions when the model is social and that interactive feedback is possible at 18 and 24 months [Nielsen, 2006, Nielsen et al., 2008].

This period of age, second half of the second year, is a crucial period when a systematic, coordinated and consistent peer to peer cooperation emerge [Eckerman and Didow, 1996]. This matches the frequency and efficiency of coordination skill : the coordinated episode are infrequent and seems more accidental at 18 months but they are frequent and effective at 24 and 30 months [Brownell and Carriger, 1990], first under simple forms (imitative game, basic routines) before being more coordinated and fully cooperative during the third year, in particular with negotiation and accomodation between participants [Brownell et al., 2006].

The precise 15 to 18 month period is emphasized as a key point, where joint engagement episodes increase greatly in frequency in free plays situations [Bakeman and Adamson, 1984]. Warneken and colleagues have focused on these episodes, and show that the trials to reengage a partner who has left the cooperative activity with an unachieved goal are eye contact for 14 month, pointing gesture for 18 months, with a verbalization in addition for 24 months old children [Warneken and Tomasello, 2007]. The emerging speech

capabilities of the children is developed between 28 and 32 months, and used to regulate cooperative activity which increases their efficiency. More precisely, the first appearance of regulatory speech occurred to negotiate the role to be played between both participants (e.g. "Give", "Go there"). Next we find verbal means to address some details, such as the timing (e.g. "Wait!") or to direct the partner's attention (e.g. "Watch!"). Eventually, the description of their own actions are verbally announced (e.g. "I get it.") [Eckerman and Didow, 1996, Eckerman and Peterman, 2001].

2.3.4 Collaborative Learning : Shared Plan

The children have now developped all the skills needed to participate in a shared cooperative activity. Indeed, three main features are mandatory in order to define such episode : a mutual responsiveness, a commitment to the joint activity and a commitment to mutual support [Bratman, 1992]. First, we have just seen that after his second year, the children is able to mediate a collaborative episode through non-verbal and verbal communicative cues [Eckerman and Didow, 1996, Eckerman and Peterman, 2001]. Second, they reengage actively the partner if he left before the plan is finished [Warneken and Tomasello, 2007]. Last, the children has also an altruistic motivation at 18 months, and spontaneously help adult to achieve the goal they are capable of understanding and representing [Warneken et al., 2006].

These features come from the fact that the children understand others as intentional agents and one uniquely human aspect is also the motivation to share these intentional states with peers [Tomasello et al., 2005]. Indeed, to have a commitment to the joint activity, the goal has to be shared by both participants, a shared (or "we") intentionality that we will do something together. Moreover, the commitment to the mutual support involves the fact that one agent has to know what the other has to do in the cooperative task, to be able to monitor the different timing or help him if a problem occurs. This could be validated if the agent are capable of role reversal, showing in this way that he is aware of all the roles to be fulfilled, including the partner's ones [Tomasello et al., 2005]. This view is summarized in the Figure 2.8 where the goal is represented with self and the other, allowing thus a shared goal : my goal is to the respect of the other goal. Another important aspect is the "bird's-eye view" of the joint intention which contain also myself and the other to allow help and role reversal, that 18 month old children can do for a role reversal imitation task [Carpenter et al., 2005].

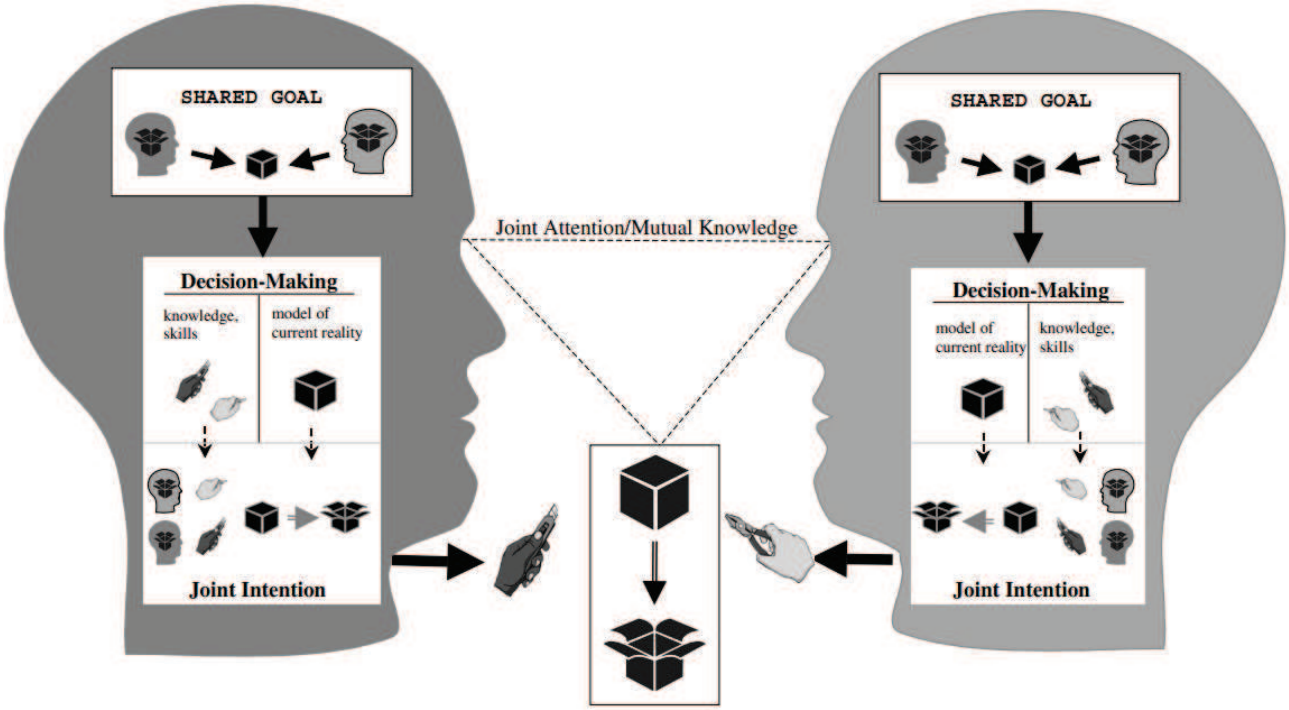


Figure 2.8: *Original Caption : Each partner's conception of a collaborative activity in which a shared goal and joint intention (with complementary roles) are formed. (From [Tomasello et al., 2005]).*

2.4 Child Development : Reasoning and Planning using Experience

With the help of adults, who could engage in direct attention to a precise element, bring knowledge by commenting or labeling an interesting focus, teach new actions with the used of imitation or cooperative scene, a children is capable of learning through social interactions with more expert caretakers or peers. One key feature which has to be developed in order to be able to use these learning methods is the intentional state that the children and the other have when they act. Agents have goals and use actions to achieve them, which means that actions have consequences and these can be our desired goals. In the same perspective, collaborative plans have shown example of actions linked together, which has to be in a precise order to be efficient, thus these goals could unlock pre-conditions needed to do a further act. This teleological stance is a powerful tool for the children to reason about his experience, and provides another support to acquire knowledge in a less explicit and caretaker-dependent method.

To be able to reason and plan, two main components are needed : knowledge data from which we could extract information, and a reasoning capability to use this knowledge and determine new concepts or information [Hayes-Roth, 1997]. We will thus investigate at first the memory capabilities of the children and how they could recall and organize their past experiences. Next we will see how inference mechanisms are recruted by the infant to learn regularities from their environment.

2.4.1 Children's Memory about their self-past experience

Indeed, one of the crucial aspect of cognition is to anticipate future events which could be achieved by humans with the ability to "travel in time", in particular by remembering past episodes and predict the next ones through imagination into the future [Vernon et al., 2007]. This ability to remember not any past events but our personal episodes used different specific memories, recording in the autobiographical memory and forming our daily life context. It is based on both the episodic memory, which stores personal experiences with specific objects and people at a precise time and place, and the semantic memory, which contains general knowledge, facts or laws about the world [Tulving et al., 1988, Conway and Pleydell-Pearce, 2000, Cohen and Conway, 2007]. Indeed, when we access to the episodic memory, we "remember" or "recollect" (e.g. "I remember having play Tic-Tac-Toe with my little brother, yesterday, in my room") whereas we "know" or "recall" something when we retrieve information from the semantic memory (e.g. "The Bastille has been taken in July 1789, the 14th during the French Revolution") [Tulving, 1972, 1989].

These 2 different types of memory have been highlighter in particular because of two different (and not present in the same patient) amnesia : temporal amnesia, a loss of memory from personal experience, and categorical amnesia, loss of acquired facts. Indeed, K.C. (Figure 2.9), after a car accident, has lost his episodic memory but not his semantic one : he could then play chess without remembering he has ever learned and played them [Tulving, 1989]. Without his episodic memory, his autonoetic consciousness (who knows himself) was removed : he could not recall his own past event, then he lives in a permanent present subjective time [Tulving, 1989, 2002]. The awareness of his personal past, to know the sequence of events which bring us in the present, creates a sense of personal history and thus defines identity and purpose [Nelson and Fivush, 2004].

Figure 2.9: *Original caption : Research involving both normal and abnormal brain activity is modifying the traditional view of memory as simply storage of information. The amnesic patient K.C. has retained his knowledge of how to play chess, although he cannot remember having played chess ever before, with anyone. The dissociation between the normal retention of knowledge and the severely impaired ability to recollect personal events suggests a distinction between two kinds of memory, semantic (involving impersonal facts) and episodic (involving personal experience). (From [Tulving, 1989]).*

A summary of these different features of semantic and episodic memory is summarized in Table 2.1.

	Episodic	*Semantic*
Type of information represented	Specific events, objects, people	General knowledge facts about the world
Type of organisation in memory	Chronological (by time) or spatial (by place)	In schemas or in categories
Source of information	Personal experience	Abstraction from repeated experience or generalisations learned from others
Focus	Subjective reality: the self	Objective reality: the world

Table 2.1: *Original caption : the episodic-semantic distinction. (From [Cohen and Conway, 2007]).*

However, these kinds of memory are interacting between each other and thus having an interdependent relationship [Cohen and Conway, 2007]. Episodic memory is accessible to inspection from the semantic memory, which could interpret data with the knowledge and modify them, what Tulving called an "encoding" process [Tulving, 1972], for when you see at first a wax statue of a french revolutionary man and you learn after that it was

in fact Napoleon : the label is then add to your remembering. On the other way around, the semantic memory could use inferential and reasoning mechanisms to extract elements from retrieval of repetitive or pertinent personal experience from the episodic memory [Tulving, 1972, Cohen and Conway, 2007].

Now that we have defined such kind of memory, we will focus on how they show up during the development of the children cognition. Indeed, memory are present in infant even before birth with newborns able to distinct familiar sounds (heard when in utero, like mother's voice uttering the same passage aloud) from novel sound [DeCasper and Spence, 1986]. The recognition capability increases to become more durable : at 6 months, children can differentiate between familiar and novel stimuli (sounds or sights) for several weeks [Fagan, 1973] and as soon as 9-months of age, infants are able to remember a sequence of actions with specific objects and could reproduce them weeks later, after presented the objects again [Bauer et al., 2000]. However, it is one thing to remember events, but one need also to be able to link past events with the present self. Inspired be the classic mirror task of Gallup [Gallup, 1970], 3, 4 and 5 years old children were recorded during a play with an experimenter covertly placed a sticker on their head and video was shown either just after or a few days later the interaction, testing their delayed self-recognition understanding. Thus, 3 year-old children never reach for the mark, and 4 year-old always do, without any regard to the proximity or delay of the sticker trick. Eventually, 5 year-old are able to reach the sticker when the video is shown just after, but not when the episode has occurred some days ago, stating that they know it was in the past and the paper is not on their head anymore, showing them that they understand the temporal relation between past and present [Povinelli et al., 1996], with a self representation linking past self to present self along his own timeline, defining his life [Fivush, 2011]. Thus this mental time travel is the last, but not the first, feature of autobiographical memory [Piolino et al., 2007].

2.4.2 Children's Teleological stance and Reasoning capabilities

Adults can use a powerful tool for their reasoning capabilities by making inferences about the cause or consequences of events, based on the observations of them and their variation or covariation [Cheng, 1997]. How and when these features emerge in children?

As we have seen before, autobiographical memories is not ready until five years, but some specific episodic memory is already in place before one year [Bauer et al., 2000]. Yet, infants are capable of inferring goals of an unanimated moving dot as soon as 12 months by taking a teleological stance [Gergely et al., 1995]. It consists of taking into account three components of the present and future : the action, the goal state (achieved in the future) and the situational constraints (in the present). Indeed, if we have access to any two of these elements, we can infer the third one, and thus obtain one of the mental state among intentions (related to the action currently executed), desires (the goal the agent wants to achieve) and the beliefs (the constraints supposed by other), as shown in figure 2.10. This reasoning is using the principle of rational actions [Gergely and Csibra, 2003], stating that :

- The purpose of an action is to bring to you the goal state in the future, when the action has been executed ;
- The goal states are achieved by an agent who select the most rational action available according to the constraints of the current situation ;

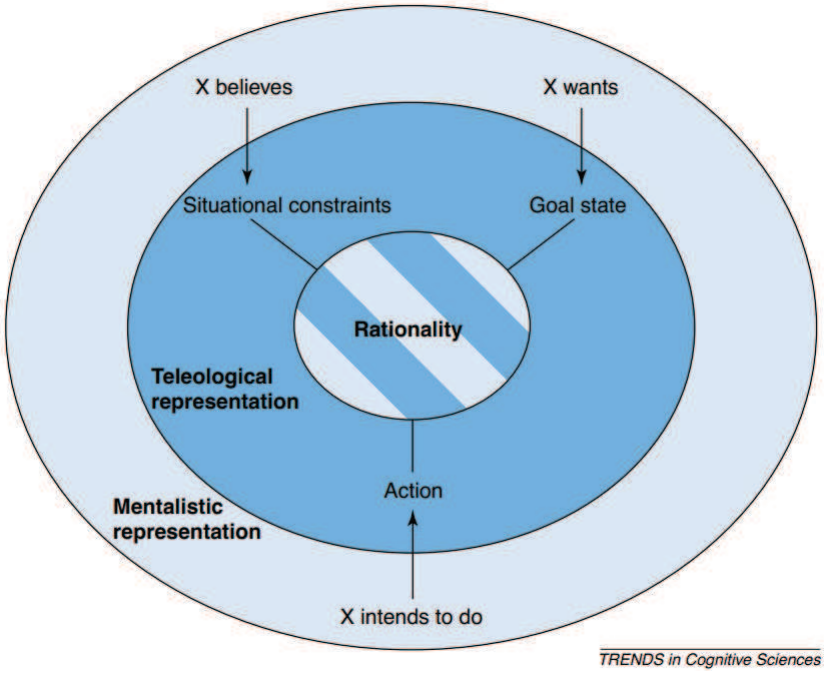


Figure 2.10: *Original Caption : Teleological and mentalistic representations of actions. Teleological representations relate three aspects of the real world to each other via the rationality principle, which provides explanations and predictions for observed actions. Mentalistic action representations involve three types of intentional mental states attributed to an agent (X). The contents of these mental states correspond to the elements of the teleological representations. There are several differences between these action explanations, including the direction of the explanation (causal versus teleological), or the ontological status of the elements (real versus fictional worlds). Note, however, that the principle of rational action applies equally to both kinds of representation. (From [Gergely and Csibra, 2003]).*

In particular, attributing goals could be used for two different process, depending on the direction of the inference : when the children determine the goal from an ongoing action ("What is the function of this action?", he could predict the future state before it actually happens, whereas anticipating the action knowing the goal ("What action would achieve that goal?") he could predict the trajectory of the move. These features are available once the children has learned the different actions to achieve these goals, especially through social learning by observing other's action and inferring their goal (thus discovering novel goals) or on the other way around, acquiring means actions in order to achieve it [Csibra and Gergely, 2007], in particular when the other is explicitly stating his action ("I am doing something") or his goal ("I want something). A summary could be found in Table 2.2.

Primary function	Type of inference	
	'Action-to-Goal'	'Goal-to-Action'
On-line Prediction	Goal prediction: Predicting the likely effect of an ongoing action	Action anticipation: Predictive tracking of dynamic actions in real time
Social Learning	Discovering novel goals and artefact functions	Acquiring novel means actions by evaluating their causal efficacy in bringing about the goal

Table 2.2: *Original Caption : The function of teleological interpretation of actions. (From [Csibra and Gergely, 2007]).*

In fact, social part seems important in the process of inferring. Indeed 24 months toddlers does not spontaneously trigger a first event to generate a second : a block moving "by itself" in a base to make a plane toy rotate on itself. However they could achieve this if an agent is moving the block or explains using causal-language [Bonawitz et al., 2010]

Part II

Publications

Chapter 3

Exploring the Acquisition and Production of Grammatical Constructions Through Human-Robot Interaction

3.1 Introduction

The following publication will detail how we have implemented in our humanoid robot iCub a system in order to acquire then produce grammatical construction, giving to the robot the capability to develop language understanding and speaking through interaction with humans. Based on the Richness of the Stimulus hypothesis ([Tomasello, 2000, Sampson, 2002, MacWhinney, 2004]), it is inspired by the strategy of the children who could grammaticaly classify word as Open Class Word or Close Class Word, in particular using prosodic cues ([Morgan and Demuth, 1996]). These classification will allows the system to use the grammatical construction theory in order to map the sentence structure to the event meaning structure ([Goldberg, 1995]) during a learning phase when the human is interaction with the robot.

3.2 Publication

frontiers in NEUROSCIENCE
IN NEUROROBOTICS

Exploring the Acquisition and Production of Grammatical Constructions Through Human-Robot Interaction with Echo State Networks

Xavier Hinaut, Maxime Petit, Grégoire Pointeau and Peter F Dominey

Journal Name:	Frontiers in Neurorobotics
ISSN:	1662-5218
Article type:	Original Research Article
First received on:	14 Jan 2014
Frontiers website link:	www.frontiersin.org

Exploring the Acquisition and Production of Grammatical Constructions Through Human-Robot Interaction with Echo State Networks

Xavier Hinaut[1,2], Maxime Petit[1,2], Gregoire Pointeau[1,2], Peter F. Dominey[1,2,3]

[1] Stem cell and Brain Research Institute, INSERM U846, 18 Avenue Doyen Lepine, 69500 Bron, France

[2] Université de Lyon, Université Lyon I, 69003, Lyon, France

[3] CNRS, France

(xavier.hinaut ; maxime.petit ; gregoire.pointeau ; peter.dominey)@inserm.fr

Abstract

One of the principal functions of human language is to allow people to coordinate joint action. This includes the description of events, requests for action, and their organization in time. A crucial component of language acquisition is learning the grammatical structures that allow the expression of such complex meaning related to physical events. The current research investigates the learning of grammatical constructions and their temporal organization in the context of human-robot physical interaction with the embodied sensorimotor humanoid platform, the iCub. We demonstrate three noteworthy phenomena. First, we demonstrate that a recurrent network model can be used in conjunction with this robotic platform to learn the mappings between grammatical forms and predicate-argument representations of meanings related to events, and the robot's execution of these events in time. Second, we demonstrate that this learning mechanism can function in the inverse sense, i.e. in a language production mode, where rather than executing commanded actions, the robot will describe the results of human generated actions. Finally, we collect data from naïve subjects who interact with the robot via spoken language, and demonstrate significant learning and generalization results. This allows us to conclude that such a neural language learning system not only helps to characterize and understand some aspects of human language acquisition, but also that it can be useful in adaptive human-robot interaction.

1. Introduction

1.1 Issues in language acquisition

The ability to learn any human language is a marvelous demonstration of adaptation. The question remains, what are the underlying mechanisms, and how do humans make the link between the form of a sentence and its meaning? Enormous debate has ensued over this question. The debate can be characterized with one end of the continuum, Piaget's constructivism, holding that language can be learned with general associative mechanisms,

and the other end, Chomsky's innatism, holding that the stimulus is so poor, that language could only be learned via a highly specialized universal grammar system (Piattelli-Palmarini 1980). We and others have argued that linguistic environment is rich – in response to the "Poverty of stimulus hypothesis" (reviewed in (Dominey & Dodane 2004)). As the child is situated in the environment, it has access to massive non-linguistic information that can aid in constraining the possible meanings of phonemes, words or sentences that it hears (Dominey & Dodane 2004). In this context, social interaction is clearly an important factor that helps the child to acquire language, by focusing its attention on the same object or event as the person he is interacting with via joint attention. Joint attention permits one to considerably reduce the possible mappings between what is said and what is happening in the environment. Joint attention happens sufficiently often to assume it as one of the reliable ways to help the child to acquire language: for instance when playing a game, showing an object, ritualized situations including bathing and feeding, etc. (Carpenter et al 1998, Dominey & Dodane 2004, Knoblich & Sebanz 2008, Ricciardelli et al 2002, Sebanz et al 2006, Tomasello 2003, Tomasello & Hamann 2012).

Despite the potential aid of joint attention, mapping the surface form onto the meaning (or deep structure) of a sentence is not an easy task. In a first step in this direction, Siskind demonstrated that simply mapping all input words to all possible referents allows a first level of word meaning to emerge via cross-situational statistics (Siskind 1996). However, simply associating words to specific actions or objects is not sufficient to take into account the argument structure of sentences in language. For instance given these two sentences "Mary hit John." and "John was hit by Mary." which have the same meaning but with a different focus or point of view, how could a purely word-based system extract the exact meaning of the sentence? How could an infant determine who is doing the action (the *agent*) and who endures the action (the *object*)? As simple this example is, relying only on the semantic words, and their order in the sentence, will not permit to reliably distinguish the *agent* from the *object*.

To begin to answer this question, we consider the notion of grammatical construction as the mapping between a sentence's form and its meaning (Goldberg 1995, Goldberg 2003). Goldberg defines constructions as "stored pairings of form and function, including morphemes, words, idioms, partially lexically filled and fully general linguistic patterns" (Goldberg 2003). Constructions are an intermediate level of meaning between the smaller constituents of a sentence (grammatical markers or words) and the full sentence itself.

Typical grammatical constructions could be used to achieve thematic role assignment, that is answering the question "Who did what to whom". This corresponds to filling in the different slots, the roles, of a basic event structure that could be expressed in a predicate form like *predicate(agent, direct object, indirect object* or *recipient)*. A simplified summary of characterization of grammatical constructions can be seen in Figure 1.

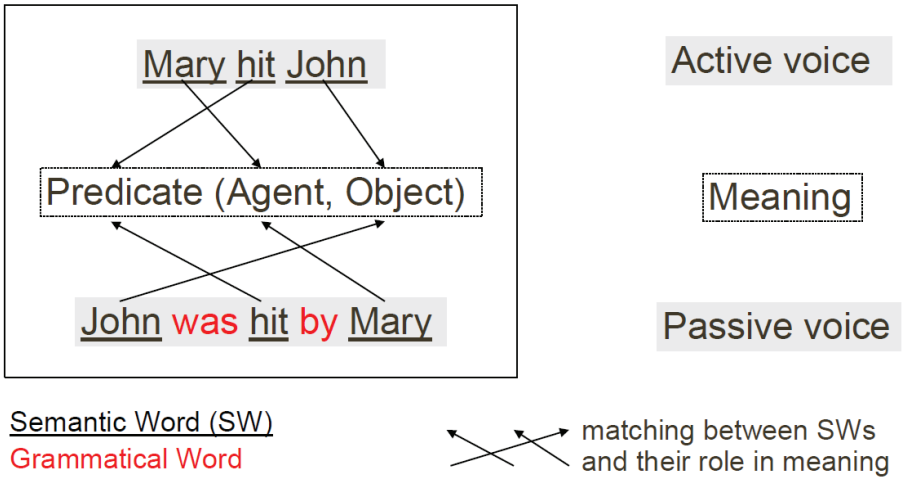


Figure 1: Schematic characterization of the thematic role assignment task. Solving this task consists in finding the adequate mapping between the content words (i.e. semantic words) and their roles in the meaning of a given sentence. This mapping is represented by the set of arrows (here three) for each sentence surface-meaning mapping.

Solving the thematic role assignment problem consists in finding the correct role for each semantic word (i.e. content word or open class word). It thus consists in finding the *predicate*, the *agent*, the *object*, and the *recipient* for a given action. In the preceding example this means that *hit* is the predicate, *Mary* is the agent and *John* is the object. How could one use grammatical constructions to solve this thematic role task for different surface forms as illustrated in Figure 1? According to the cue competition hypothesis of Bates and MacWhinney (Bates & MacWhinney 1987, Bates et al 1982) the identification of distinct grammatical structures is based on combinations of cues including grammatical words (i.e. function words, or closed class words), grammatical morphemes, word order and prosody. Thus the mapping between a given sentence and its meaning could rely on the ordered pattern of words, and particularly on the pattern of function words and markers (Dominey 2003, Dominey et al 2003). As we will see in the Material and Method section, this is the assumption we make in the model in order to resolve the thematic role assignment task, that

is, binding the sentence surface to its meaning. In English, function words include “the”, “by”, “to” ; grammatical markers include verb inflexions “-ing”, “-ed” or “-s”. One interesting aspect of grammatical words and markers is that there are relatively few of them, compared to the potentially infinite number of content words (i.e. semantic words). Hence the terms “closed class” for grammatical words and “open class” for semantic words. As these closed class words are not numerous and are often used in language, it could be hypothesized that children would learn to recognize them very quickly only based on statistical speech processing. This argument is reinforced by the fact that such words or markers are generally shorter (in number of phonemes) than content words. This notion of prosodic bootstrapping (Morgan & Demuth 1996) is reviewed and modeled in Blanc et al. 2003 (Blanc et al 2003).

1.1 Overview of the tasks

In this study we investigate how a humanoid robot can learn grammatical constructions by interacting with humans, with only a small prior knowledge of the language. This includes having a basic joint attention mechanism that allows the robot to know for instance what is the object of focus. We approach our simplified study of language acquisition via two conditions: language comprehension and language production. Both conditions will have two modes: a training mode, when the human acts as a kind of teacher, and a testing mode, where the human could test the language capabilities of the robot as in child-caregiver interactions. The experimental tasks will test the ability of our neural network model of language acquisition to understand and to produce meaningful language.

We have shown in previous studies that the neural model used (1) can learn grammatical constructions correctly generated with a context-free grammar (with one main and one relative clause), (2) can show interesting performance in generalizing to not learned constructions, (3) can show predictive activity during the parsing of a sentence and in some cases give the final correct parse before the sentence ended, and (4) that the neural activity may be related to neurophysiological human recording (Hinaut & Dominey 2012, Hinaut & Dominey 2013). We believe that these results demonstrate that the model may be suitable to a developmental robotic approach, extending our previous work in this domain (Dominey & Boucher 2005a, Dominey & Boucher 2005b).

Here we have four goals: (1) to determine if it is possible to use the model in an interactive fashion with humans, that is, to integrate this neural model in the robotic architecture and

make it communicate and work in real-time with the other components of the architecture (speech recognition tool, etc.); (2) test the model in a productive manner, that is instead of "understanding" a sentence, it will be able to produce one, that is, to produce the sequence of words of the grammatical structure given the thematic roles and the sentence type (canonical or non-canonical); this has not been done in our previous experiments with the neural model; (3) in the comprehension task, test if the neural model can learn constructions that allow for commands that manipulate the temporal structure of multiple events. For instance to correctly respond to the sentence "before you put the guitar on the left put the trumpet on the right". Finally, (4) we test the model with language input from naïve subjects, in order to determine if indeed this adaptive approach is potentially feasible in less structured environments.

In the Material and Methods section we will first briefly present the robotic platform and the interaction environment. We will then describe the two neural models used for the comprehension and production tasks. Finally, the integration of these components will be presented. In the Experiment section we will describe the experimental procedures for the *scene describer* task, and the *action performer* task. In Results section we will illustrate the functioning of the system in these two modalities, including figures illustrating the human-robot interactions, and figures illustrating typical neural activation recorded for both models. We then present the data and learning and generalization results for an extended experiment with 5 naïve subjects. In the last section, we will discuss the results and interesting aspects that the combination of a comprehension and production neural models provide. Training and testing data used in the experiments, and corresponding to the figures showing the output neural activity of the models are provided in Appendices section.

2. Material and Methods

2.1 iCub platform and interaction architecture

The platform that we used is the iCub, furnished by the FP6 EU consortium RobotCub (see Figure 2). The iCub (Metta et al 2010) is a 53 DOF humanoid robot built by the Italian Institute of Technology (IIT) with the size of a three and a half year-old child. We use YARP (Metta et al 2006) as the robotic middleware with the Biomimetic Architecture for Situated Social Intelligence Systems (BASSIS architecture) built for the FP7 Experimental and Functional Android Assistant project (Petit et al 2013).

The Supervisor module is implemented with the CSLU RAD Toolkit (Sutton et al 1998) Rapid Application Development for spoken language interaction. It uses the Festival system

for speech synthesis (Taylor et al 1998) and Sphinx II for spoken language recognition (Huang et al 1993). The Supervisor provides a dialog management capability built as a finite-state system. This capability allows the user to guide the robot into the different states of behavior, but is distinct from the neural language model, described below. The Supervisor/Manager orchestrates the communication and exchange of data between speech recognition and synthesis, the neural models for language comprehension and generation, and the robot perception and action systems.

The ability of the iCub to perceive physical objects and their manipulation in the context of action performance and description is provided by the ReacTable, which detects objects on a translucid table based on detection of fiducial markers on the object bases, using an infra-red camera (Bencina et al 2005). The ReacTable thus provides data on the type and position of objects on the table with high precision. The ReacTable is calibrated into the motor space of the iCub, so that object locations can be used for physical interaction.

The motor control for iCub reaching, grasping and object manipulation is provided by DForC – Dynamic Force Field Controller – (Gori et al 2012), based upon dynamic force control. The robot has a small set of primitive actions: put(object, location), grasp(object), point(object).

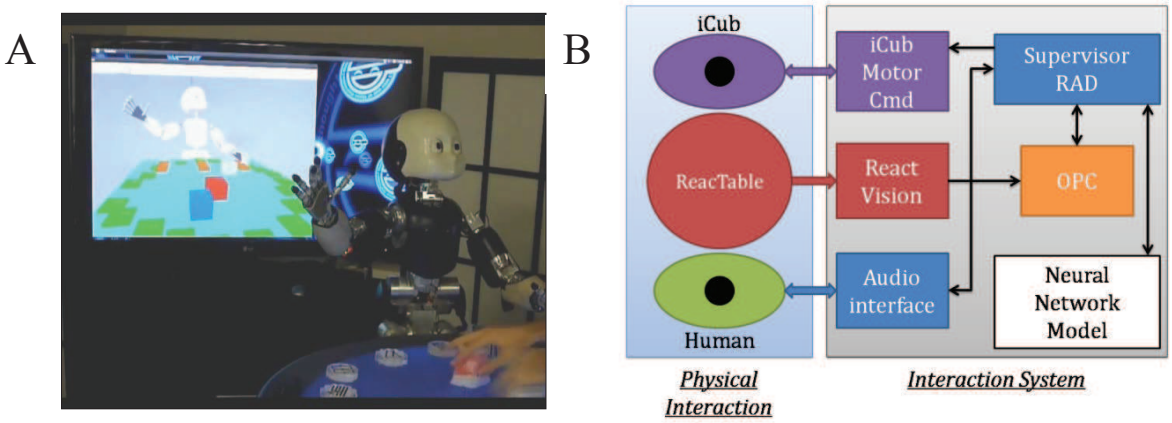


Figure 2: Robotic Platform. (A) iCub humanoid robot with the ReacTable. (B) System architecture overview. The Supervisor coordinates all interactions between the human and the different components of the system. When the human moves an object on the ReacTable, the coordinates are transformed into the robot space, and stored in the Object Properties Collector (OPC). For Action Performance when the human speaks, the words are recognized by the audio interface, then they are packaged and sent to the Neural Network by the Supervisor. Resulting commands from the Neural Network are processed and forwarded to the iCub Motor Command (iCub Motor Cmd) interface by the Supervisor, the robot then performs the given actions. For Scene Description, the Cartesian coordinates of the objects are transmitted from the OPC to the Supervisor. Spatial relations between "environmental" objects and the object of focus are computed. They are then sent to the Neural Network together

with the sentence type (canonical or non-canonical). The sentence generated by the Neural Network is sent to the Audio interface for speech synthesis, again under the control of the Supervisor.

2.2 Neural language model

The neural language processing model represents the continued development of our work based on the underlying concept of a recurrent network with modifiable readout connections for grammatical construction processing (Dominey 2003, Dominey et al 2003, Hinaut & Dominey 2012, Hinaut & Dominey 2013). As described in the context of grammatical constructions above, for sentence processing we have shown that the pattern of open and closed class word order could be used to correctly identify distinct grammatical constructions and extract their meaning for a small set of sentences. More recently we have demonstrated the extension of this ability to larger corpora from several hundreds of uniquely defined construction-meaning pairs, to tens of thousands distinct constructions including redundant and ambiguous meanings (Hinaut & Dominey 2013). As the neural model has anytime learning property, it is of interest to use it for exploring language acquisition in a developmental robotics perspective.

The core of the language model is a recurrent neural network, with fixed random connections, which encodes the spatio-temporal context of input sequences. This sequence-dependent activity then projects via modifiable connections to the read-out layer. Modification of these read-connections by learning allows the system to learn arbitrary functions based on the sequential input. This framework has been characterized as Reservoir Computing (Lukosevicius & Jaeger 2009, Verstraeten et al 2007), where the recurrent network corresponds to the reservoir, and has been developed in different contexts. The first expression of the reservoir property with fixed recurrent connections and modifiable readout connections, was developed in the context of primate neurophysiology, with the prefrontal cortex as the reservoir, and modifiable cortico-striatal connections as the modifiable readout (Dominey 1995, Dominey et al 1995). Further development was realized in related systems including the Liquid State Machine (Maass et al 2002), and Echo State Network (Jaeger 2001, Jaeger & Haas 2004).

The model employed in the current research (Hinaut & Dominey 2013) pursues this parallel between brain anatomy and the reservoir computing framework. Prefrontal cortex is modeled as a recurrent network that generates dynamic representations of the input, and striatum as a separate population connected to cortex via modifiable synapses, which learns to

link this dynamic representation with a pertinent output. Cortex and striatum corresponding respectively to the reservoir and readout. The reservoir is composed of leaky neurons with sigmoid activation. The following equation describes the internal update of activity in the reservoir:

$$x(t+1) = (1-\alpha)x(t) + \alpha f(W_{res}\, x(t) + W_{in}\, u(t+1)) \quad (1)$$

where x(t) represents the reservoir state; u(t) denotes the input at time t; α is the leak rate; and f(·) is the hyperbolic tangent (tanh) activation function. W*in* is the connection weight matrix from inputs to the reservoir and W*res* represents the recurrent connections between internal units of the reservoir. In the initial state, the activation of all internal units of the reservoir is zero. The inverse of the leak rate (1/α) could be interpreted as the time constant of the system.

By definition, the matrices W*in* and W*res* are fixed and randomly generated. Internal weights (W*res*) are drawn from a normal distribution with mean 0 and standard deviation 1 and then rescaled to the specified spectral radius (the largest absolute eigenvalue of the W*res* matrix). The input weight matrix W*in* was first generated with values chosen randomly between -1 and 1 with a 50% probability. The W*in* matrix was then rescaled depending on the experiment (*input scaling* parameter). The density of the input connections is 100%.

The output vector of the system which models the striatum is called the readout. Its activity is expressed by the following equation:

$$y(t) = W_{out}\, x(t) \quad (2)$$

with W*out* the matrix of weights from the reservoir to the readout (output). The activation function of readout units is linear. Interestingly, the readout activity gives a pseudo-probabilistic response for each output unit. To train the read-out layer (i.e. compute W*out*), we use a linear regression with bias and pseudo-inverse method (Herbert Jaeger, 2001). This general model is applied in two distinct instantiations. One model processes commands (sentences) and generates a predicate-argument representation of the meaning. The second describes observed actions, i.e. given a predicate-argument meaning as input, it generates a sentence describing that meaning. Thus, the comprehension system learns to map semantic words of input sentences onto an output that characterizes the role (action, agent, object, recipient) of each of these semantic words, based on the structure of grammatical words in the sentence. The production system learns the inverse mapping, from the meaning (i.e. specification of the role of each semantic word) onto a sentence form.

2.2.1 Comprehension model for Action Performing task

The architecture of the comprehension model is illustrated in Figure 3.

Preprocessing: Before being provided as input to the neural model, the sentence must first be transformed by extracting the open-class (i.e. semantic) words. The resulting grammatical form is characterized by the sequential pattern of closed-class (i.e. grammatical) words. This operation is performed by replacing all open class words by 'SW' markers (SW: semantic word). The semantic words removed from the sentence are stored in a working memory. The working memory acts as a first-in-first-out (FIFO) stack: the words will be retrieved in the same order as in the output. For example, when semantic word 2 (SW2) is determined by the model to be the agent, the actual word corresponding to SW2 will be retrieved as the agent of the described action. The closed class words used were: 'after', 'and', 'before', 'it', 'on', 'the', 'then', 'you'.

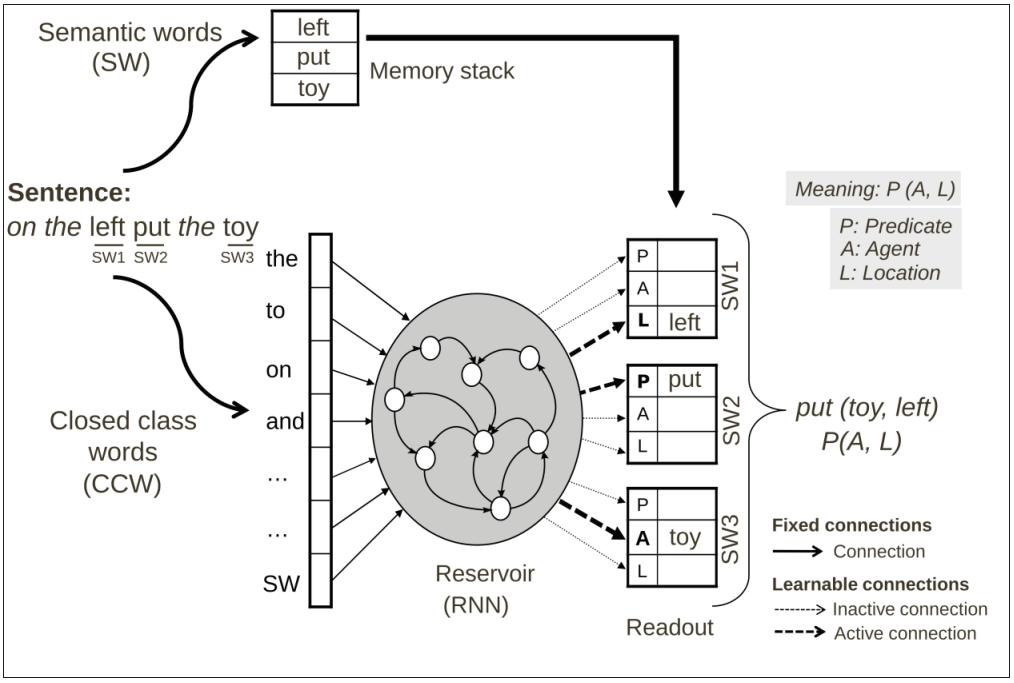


Figure 3: Neural comprehension model for the Action Performing task. Sentences spoken by the user are first transformed into grammatical forms, i.e. all semantic words (SW) are replaced by a *SW* marker. The reservoir is given the grammatical form word by word. Each word activates a different input unit. Based on training, the readout connections from the reservoir provide the coding of the predicate-argument meaning in the readout neurons, thus forming the grammatical construction as a mapping from grammatical form to meaning. The meaning of an input sentence is specified by determine the role (predicate, agent or location) for each semantic word SW.

Reservoir parameters: The number of unit used in the reservoir is 100. The leak rate used is 1/6 (=0.1666…). The *input scaling* is 0.75. The *spectral radius* is set to 1.

Sentence input parameters: Given an input sentence, the model should assign appropriate thematic roles to each semantic word. The presentation of inputs is illustrated in Figure 3. Sentences are represented in the input as grammatical forms, where specific instances of noun and verb words (semantic words – SW) are replaced by a 'SW' marker. Thus, a given grammatical construction can code for multiple sentences, simply by filling in the 'SW' markers with specific words. In this way of coding, the reservoir cannot distinguish between nouns or verbs, as they have the same input neuron. This is an interesting characteristic when using the model within a robotic platform, because when sentences are processed there is no need to do a preprocessing in order to classify words as nouns or verbs.

The total number of input dimension is 9; 8 for closed class words, 1 for the semantic word marker. Each word is coded as a square wave of 1 time step. There is no pause between successive word presentations (the use of pauses does not have significant influence on the results), but there is a final pause at the end of the sentence in order to inform the model that the sentence is finished. This final pause could be replaced by a period, as it would have the same function as a terminal symbol. An offset of the sentence was added at the beginning of the inputs if they were not of maximal length, in this way the correct final meaning is always given at the last time step.

Desired meaning output coding: Making the analogy with an infant who is learning language in the presence of sentences and their corresponding meanings, we consider that the system is exposed to a meaningful scene while the input sentence is being presented. Thus, the system has access to the meaning starting at the beginning of the presentation of the sentence, hence the desired output teacher signal is provided from the beginning of the first word until the end of the input. All the output neurons coding the meaning are clamped at 1, all other output neurons are clamped to 0. By forcing the correct outputs to be 1 from the onset of the sentence during learning, we obtain predictive activation when processing (i.e. testing) a sentence after the learning phase. This can be seen in the results section in Figure 8, below (see (Hinaut & Dominey 2011, Hinaut & Dominey 2013) for more details). The meaning output dimension is 36 (=6*3*2): 6 semantic words that each could have 3 possible thematic role assignment (predicate, agent or location), for each of up to maximum 2 verbs.

Post processing: To determine the meaning specified in the output, the activity of the output at the last time step is thresholded. For each SW, we take the role that has the

maximum activation (if there are several). Each semantic word in the FIFO stack is then bound with its corresponding role(s). The full predicative meaning is then obtained and written in the output data file in order to be processed by the Supervisor module, and then used to command the robot.

2.2.2 Production model for Scene Description task

We have described the functioning of the language model that learns to map input sentences onto a predicate-argument representation of their meaning. Now we consider the reverse case, where given a meaning, the model should produce a sentence. This model thus employs the same principals as the language comprehension model, but we now perform the reverse operation - from a meaning we want to generate the corresponding sentence (see Figure 4). It is important to recall that there are potentially multiple possible sentences for describing a given scene or meaning (as illustrated in Figure 1). To resolve this ambiguity, we provide additional input to the model, to indicate if we want a canonical (e.g. standard, active voice) or a non-canonical (e.g. passive voice).

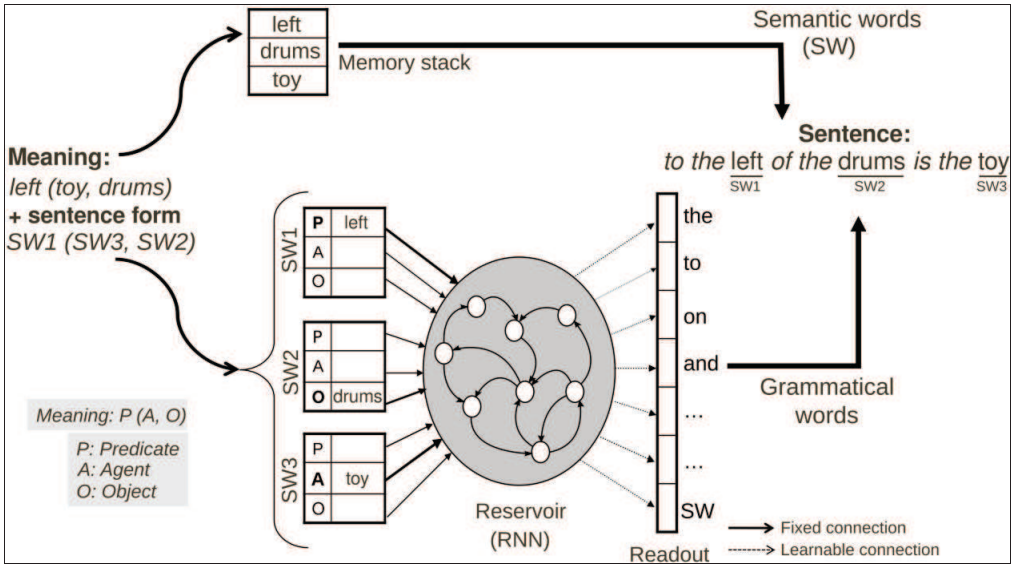


Figure 4: Neural production model for Scene Description task. The input has 2 components: (1) meaning format {Predicate(Agent, Object) - left(toy, drums)} corresponding to relation *toy to the left of drums*, and (2) construction format with {SW1 – Predicate, SW2 – Object, SW3 - Agent} which could be written in a compact way as SW1(SW3, SW2). The full input information could be represented as {SW1_Predicate – left}, {SW2_Object – drums}, and {SW3_Agent – toy}. The system must find a construction that allows this mapping of SWs to thematic roles. SW#_θ: Semantic Word # has thematic role θ, with # the relative position in the sentence among all Semantic Words.

Preprocessing: The model is given the meaning and the sentence type desired (canonical or non-canonical) by the Supervisor module. This information is converted in the corresponding coded meaning, as described in Figure 4. The semantic words of the meaning are stored in the FIFO memory.

Reservoir parameters: The number of units used in the reservoir is 500. The leak rate used is 0.75. The *input scaling* is set to 0.01. The *spectral radius* is set to 2.

Input and output coding: The coded meaning is given, for all the input units concerned, as a constant input activation set to 1. Remaining input units are set to 0. This is consistent with the output representation of the meaning in the first model presented in 2.2.1 (comprehension model). As illustrated in Figure 4 the desired mapping of the open class words onto thematic roles is specified by activating the appropriate input neurons. The input activation lasts during all the input presentation. The input dimension is the same as the output dimension of the comprehension model 6*3*2=36: 6 semantic words that each could have 3 possible thematic role assignment (predicate, agent or object), and each could have a role with at maximum 2 verbs. Table 1 illustrates how different coded-meanings can be specified for the same input meanings. This allows us to specify in the input if the sentence should be of a canonical or non-canonical form.

	Meaning	**Sentence**	**Coded-meaning**
Canonical	left(toy, drum)	The toy is left of the drums	SW2(SW1, SW3)
Non-Canonical	left(toy, drum)	To the left of the drums is the toy	SW1(SW3, SW2)
Double Canonical	left(violin, trumpet); right(violin, trumpet)	The violin is to the left of the trumpet and to the right of the guitar	SW2(SW1,SW3); SW4(SW1, SW5)
Double Non-Canonical	left(violin, trumpet); right (violin, guitar)	To the left of the trumpet and to the right of the guitar is the violin	SW1(SW5,SW2); SW3(SW5, SW4)

Table 1. Representation and form of canonical and non-canonical sentences. Both examples of each single or double type have the same meaning. The sentences are different, and the mapping of semantic words onto the thematic roles in the meaning is different, as specified in the coded-meaning or sentence form. The semantic word that is the grammatical focus changes between canonical and non-canonical sentences. Both *Meaning* and *Coded-meaning* use the convention Predicate(Agent, Object). SW#: Semantic Word #, with # the relative position in the sentence among all Semantic Words.

Activation of the output units corresponds to the successive words in the retrieved construction. The closed class words used were: 'and', 'is', 'of', 'the', 'to', '.' (dot). The dot is optional and was not used for the experiments shown in Figure 9; it could be used in the future if several sentences have to be produced. The total number of output dimension is 7: 6 for closed class words and one for the SW marker.

The output teacher signal is as the following: each word is coded as a square wave of 5 time steps. Each word was separated with a pause of 5 time step. We used 5 time steps for each word and a pause of same duration between them in order to have an output activity that last a sufficiently long time; in this way each word could be discriminated more easily in the post-processing process. There is a final pause at the end of the teacher signal. All the teacher signals were of maximal length corresponding to the longest sentence.

Post processing: Once again, the output activity is first thresholded. Then each time an output exceeds the threshold, the corresponding word is added to the final construction (if the activity of this word last 4 or 5 time steps above the word it is considered only once). If several outputs are above the threshold, the word of maximal value is kept. Finally, the sentence is reconstructed replacing the SW makers with the semantic words kept in memory.

2.3 Integrated System

The system operates in real-time in a human-robot interaction. Figure 5 shows how the communication between modules is performed. Again, the system can operate in "action performer" (AP) and in "scene description" (SD) tasks, and the Supervisor module allows the user to specify which of these tasks will be used. The Supervisor interacts with the human through spoken language to determine if he wants to run the system in train mode – to teach the robot new <meaning, sentence> pairings – or in test mode – to use the corpus of pairings already learned by the robot. Thus there are two tasks (AP or SD), each of which can be run in two execution modes (train or test). Details for AP and SD tasks are provided in the next section. Now we briefly describe train and test modes.

In train mode, the Supervisor incrementally generates one of the two training data files depending on the task (AP or SD). The human speech is transformed into text via the speech-to-text tool, and the meaning is given by the robotic platform (from perception or action). The <meaning, sentence> pairing is then written in the training data text file. In order to avoid populating the training files with bad examples in case of incorrect speech recognition, before

writing the file the Supervisor asks the user for a verification (e.g. if it correctly understood the meaning). If the user wants the example to be added to the data file he answers "yes", otherwise he answers "no".

In test mode, the Supervisor processes the test example given by the user: in AP task the example is a sentence; in the SD task the example is a meaning (i.e. the user places objects in particular positions relative to the object of focus). This test example is a half-pairing of a complete sentence-meaning pair. First, the Supervisor generates a file containing the previously established training data, and the test example. It then launches the corresponding neural model (comprehension or production) depending on the task (respectively AP or SD). The neural model is trained with the training data, and then it processes the test half-pairing and generates the "missing half" in a text file. The Supervisor processes the file returned by the neural model and executes the action in the AP task or produces the sentence in the SD task.

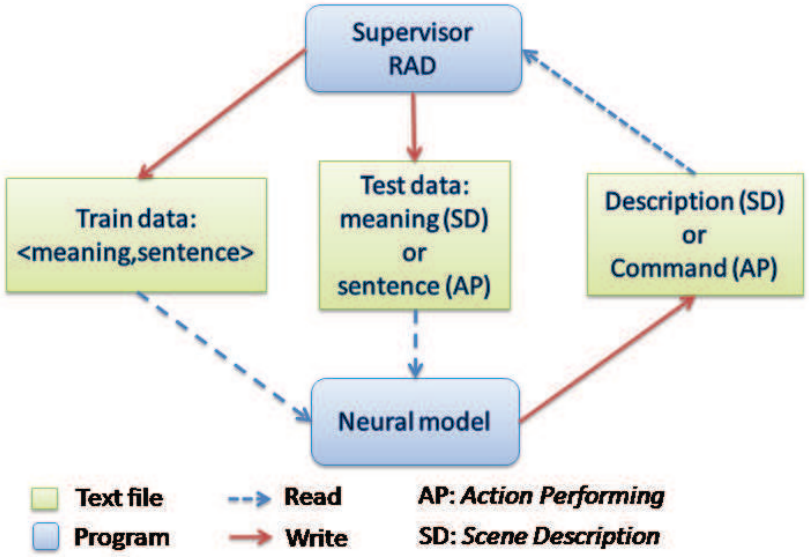


Figure 5: Communication between modules. The Supervisor manages the spoken interaction with the user and controls the robotic platform, providing different behaviors in SD and AP tasks. Depending on the mode selected by the user, train or test, it launches the neural model or not. In the train mode, pairs of <meaning, sentence> are stored in the train data file. In test mode, the sentence to be tested is written in the test data file, and both train and test files are sent at once to the Neural model. See Figure 2 for complementary information.

3. Experiments

We now illustrate in detail how the system works in two distinct modes: training and testing for the AP and SD tasks. An overview is provided in Table 1. In both tasks, meanings are expressed in a predicate-argument form: for instance *put(toy, left)* (for Action Performing task; see Figure 3), or *left(toy, drums)* (for the Scene Description task; see Figure 4). During training, meaning is produced by transforming the events and relative position of objects into the respective action and scene meanings. This is achieved by analyzing the change in object positions on the ReacTable (in order to get scene meanings) and by interrogating the program generating random robot action (for action meanings). Spoken sentences are transformed from a speech record into a list of words (using the Sphinx II recognizer) and paired with the associated meaning to populate the training database. The training mode is responsible for building a corpus of <sentence, meaning> pairs which will be fed to the neural model in order to train it. The human is then invited to build the database by interacting with the robotic platform. The type of interaction is different according to the task, AP or SD, as indicated in Table 2. In testing mode, the human provides one component of a <sentence, meaning> pair, and gets the missing component of the pair in return.

	Action Performer (AP)	**Scene Describer (SD)**
Training	1. Robot generates random action(s) [meaning] 2. Human says a corresponding command [sentence]	1. Human arranges objects on the table [meaning] 2. Human describes the scene [sentence]
Testing	1. Human says a command [sentence] 2. Robot performs corresponding action(s) [meaning]	1. Human arranges objects on the table [meaning] 2. Robot describes the scene [sentence]

Table 2. Summary of events in Training and Testing modes for the Action Performer (AP) and Scene Describer (SD) tasks. In brackets is indicated the half-pairing generated corresponding to each event.

3.1 Experiment Scenario 1: Action Performing task

In the following X, Y and Z are arbitrary objects (e.g. guitar, trumpet, violin), and, L and R are different locations (e.g. left, right, middle). In the training mode, one or two random action(s) are generated by the iCub using available objects (e.g. <put X on the R>, <grasp Y, point Z>, …). This produces the *meaning*. At the same time, the human user observes and then says the order (i.e. command) which, according to him, should command the robot to perform the(se) action(s): this corresponds to the *sentence*. The <*sentence, meaning*> pair can thus be constructed. The robot continues to randomly select possible actions and execute them, and the user provides the corresponding command, thus populating the database with <sentence, meaning> pairs.

In testing mode, the system uses the data generated in the learning mode in order to fully interact with the human, whereas in the training mode the system is more passive. In the Action Performing task the human says a command to the robot (providing the *sentence*). This test sample is passed to the neural model (Figure 3). The neural model produces the corresponding *meaning*, which is sent back to the Supervisor which translates the meaning into the corresponding robot command(s). The robot then produces the desired action(s).

3.2 Experiment Scenario 2: Scene Description task

During the training phase for Scene Description task the user puts several objects on the table and specifies the focus object. Then he describes orally one or two spatial relations relative to the focus object (e.g. <the X is to the L of Y and to the R of Z>, …), providing the *sentence*. The Supervisor then uses the coordinates of the objects and the knowledge of the focus objects to find the relationships between the focus element and the other element(s) on the table, providing the *meaning*.

During the testing phase for the Scene Description task the user puts some objects on the table in a particular spatial relation, producing the *meaning*. This test example is passed to the neural model. The latter produces the corresponding *sentence* that is sent back to the Supervisor which produces the sentence via the audio interface (text-to-speech tool).

For both tasks during testing phase the data file that is transmitted to the neural model contains both the testing data and the training data. This permits to avoid executing the neural

model each time one example is learned. Thus the model learns the whole data set and then applies this to the test data (on which it is not trained).

3.3 Experiment Scenario 3: Naïve Subject Action Performer task

In order to test the robustness of the system, we tested learning and generalization with data produced by 5 naïve subjects. In order to standardize the experiment we made a movie of a human performing a set of behaviors: 5 single actions and 33 double actions. For instance {*point(guitar)}* is an example of a single action: a corresponding sentence could be "Point to the guitar"; And {*point(guitar), put(toy, left)}* is an example of a double action: a corresponding sentence could be "Point to the guitar then put the toy on the left". For each behavior (i.e. for each scene of the movie), we asked the subjects to give a "simple" command, and then a more "elaborate" one corresponding to the observed action(s), as if they wanted a robot to perform the same action. The subjects looked at the same scene twice, once before giving a "simple" command (i.e. order), and once before giving an "elaborate" one. Subjects saw each scene twice in order to obtain more spontaneous responses from them. Thus subjects do not have to remember the scene and try to formulate both simple and elaborate sentences in a row. This resulted in a corpus of 5 (subjects) x 38 (behaviors) x 2 (canonical and non-canonical) = 380 sentences. The <sentence, meaning> corpus was obtained by joining the corresponding meanings to these sentences. Once this corpus was obtained, first, in order to assess the "learnability" of the whole corpus, we trained and tested the neural model using the same data set. Then generalization capability was tested using leaving-one-out method (i.e. cross validation with as many folds as data examples): for each <sentence, meaning> pair, the model was trained on the rest of the corpus, and then tested on the removed <sentence, meaning> pair.

4. Results

4.1 Human robot interaction

The iCub robot learns in real-time from human demonstration. This allows the robot to (1) perform complex actions requested by the user, and (2) describe complex scenes. Here "complex" means multiple actions with temporal (chronological) relations. The system can for instance execute commands like: "Before you put the guitar on the left put the trumpet on the right." We demonstrate how this form of temporally structured grammatical construction can be learned and used in the context of human-robot cooperation.

In Figures 6 and 7, we can see images extracted during human-robot interactions for the two tasks. In Figure 6, the robot is performing the motor commands corresponding to the sentence "Point the guitar before you put on the left the violin." (A) the robot is pointing the "guitar" (blue object), (B) the robot is finishing the displacement of the "violin" (red object). In Figure 7, the robot has to describe the scene relative to the object of focus: (A) the user sets the object of focus in the scene, where other objects are already present; (B) the robot is describing the position of the focus object relative to the other objects.

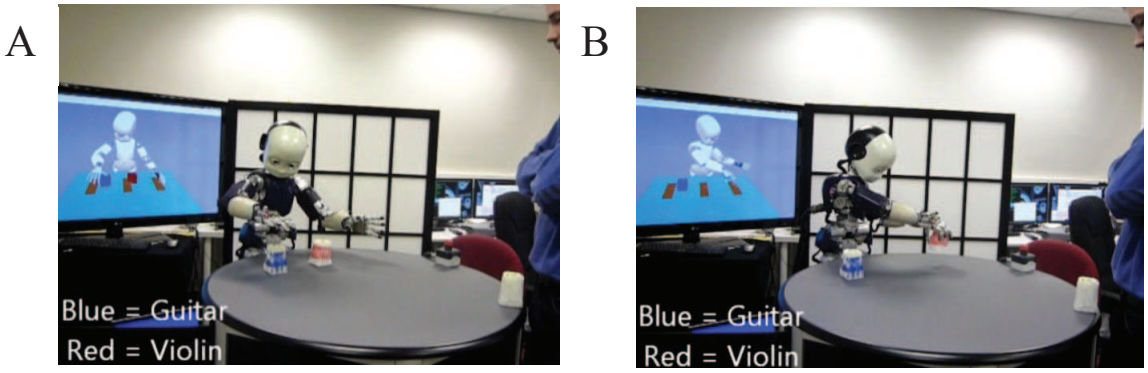


Figure 6. Action Performing task. The robot is performing the motor commands corresponding to the sentence "Point the guitar before you put on the left the violin.": (A) the robot is pointing "guitar" (blue object), (B) the robot is finishing the displacement of the "violin" (red object).

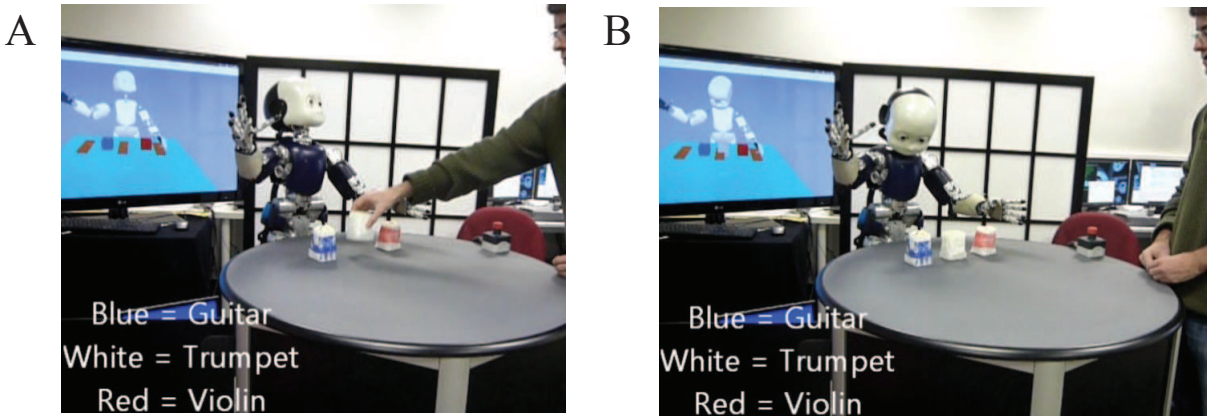


Figure 7: "Scene Description" condition. The robot have to describe the scene relative to the object of focus: (A) the user sets the object of focus in the scene, where other objects are already present; (B) the robot is describing the position of the focus object relative to the other objects.

In the following subsections we describe events and human-robot interactions during testing mode. These descriptions correspond to the transcript of sections of videos obtained

during experiments with the Action Performer and Scene Describer tasks. Videos are available in the supplementary materials.

4.2 Trials for Execution of Action Performing task

For illustrating the Action Performer task, we will use the same initial positions for the objects across trials, and vary the oral description made by the user. The initial positions of objects are the following: the guitar is on the right and the violin is on the left. Absolute and relative positions are defined relative to the user. For the first trial, the action the user asked the iCub to perform was *put (violin, left)*. The user told the robot the non-canonical sentence 1 in Table 3. The robot then put the violin on the left. For trials 2, 3 and 4, the desired first action was *point (guitar)* and the second action was *put (violin, left)*. The robot first pointed to the guitar and then put the violin on the left. In this way, we could test for different ways of saying the same meaning but with different grammatical forms of sentences (i.e. with different grammatical constructions). For each sentence spoken to the robot, we indicate in Table 3 for clarification the type of sentence (canonical or non-canonical word order) and the order of actions (chronological or non-chronological order of actions).

#	**Sentence**	**Type**
1	on the left put the violin	non-canonical
2	after you point the guitar put the violin on the left	canonical, chronological
3	before you put the violin on the left point the guitar	canonical, non chronological
4	point the guitar before you put on the left the violin	canonical & non-canonical, chronological

Table 3. Example of sentences used during testing mode of Action Performer task.

4.3 Trials for Scene Description task

For the Scene Describer task, the user was first asked to set the initial position of the objects (this is indicated by the robot when it says “set objects”). Then the user was asked to identify the focus object (indicated by “focus”). The latter is what should be in focus in the sentence, i.e. the subject (i.e. the agent in this case) of the sentence. After that, the robot asked the user if it should produce a usual (i.e. canonical) or non-canonical sentence. Then

the robot processed all the information and described the scene by producing a sentence ("robot sentence"). See Table 4 to see a subset of trials for the Scene Describer task.

Trial	Set objects	Focus	Sentence Type	Robot Sentence
1	guitar middle	trumpet right	canonical	The trumpet is to the right of the guitar
2	trumpet middle	guitar left	non-canonical	To the left of the trumpet is the guitar
3	guitar left, trumpet right	violin middle	canonical	The violin is to the right of the guitar and to the left of the trumpet
4	guitar left, violin right	trumpet middle	non-canonical	To the left of the violin and to the right of the guitar is the trumpet

Table 4. Subset of trials for the Scene Describer task. *Set objects* indicates the position of initial object(s) on the tactile table. *Focus* indicates the object that is put on the table when the "focus object" is asked by the robot. *Sentence-type* indicates the type of sentence that should be generated. *Robot Sentence* indicates the corresponding sentences produced by the robot.

In order to get an appreciation for the near real-time behavior of the system, we examined experimental data logs and collected data from 22 experiments with the scene describer and from 66 experiments with the action performer.

The execution times for the Scene Describer task are recorded from when the subject places the objects on the table, until the system responds with the description. This includes file transfer time from the Supervisor to the neural network model, and back, along with the model processing. Despite these delays, the total time of execution is around 30 seconds, which is near-real time performance. Likewise, for the action performer, processing of the spoken sentence by the model takes place within approximately 20 seconds, and then the execution of the actions by the robot is slower. This long time for executing actions is due to (a) safety limits on velocity, and the fact that (b) many of the commanded actions include two distinct actions. Still, from spoken command to completed action, the execution is less than a minute, again, within the bounds of near-real time performance.

Looking in more detail at the time used by actually running the neural network, we measured the time from sending the file to the network, to the time to retrieve the file containing the actions to be sent to the robot. For 66 trials of the AP task this required on average 6.02 seconds (SD $\pm$ 0.33 sec), and for 22 trials of the SD task the file transfer and neural network execution required 9.42 seconds (SD $\pm$ 0.96 sec). This can be considerably

improved by replacing the file-based communication with a client-server communication in the YARP framework.

4.4 Neural output activity of the models

In this section we will illustrate the activity of the neural network model for the two tasks. One has to recall that the output of the neural network is used to generate the behavioral and spoken responses.

4.4.1 Comprehension model neural activity for Action Performer task

In Figure 8 we illustrate the output activity for two example trials on the Action Performer task. From the beginning of the input of the grammatical construction the read-out activity starts to change and is updated each time a new word is presented in input. This activity can be interpreted as an estimated prediction given the inputs thus far. These estimations are based on the statistics of the sentence forms of the training corpus (see (X. Hinaut & Dominey, 2013) for details). In Figure 8A, the model correctly determines that there is only one meaning-predicate which is *put (trumpet, left)*. We see that at the last time step the neural activations concerning the on-going predictions on a potential 2nd predicate-meaning all fall below the threshold of 0.5, and as a consequence only one predicate-meaning is considered.

In some cases, this activity can be used to know the correct response before the end of the sentence. In future experiments, this could potentially allow the robot to start moving the object before the end of the sentence. This is actually a behavior that seems natural in human interaction when one give the other a series of orders. When the first order is given the human can start to do the 1st action while listening to the rest of the orders (for instance when someone lists what has to be done for a cake recipe, while another one is making the cake).

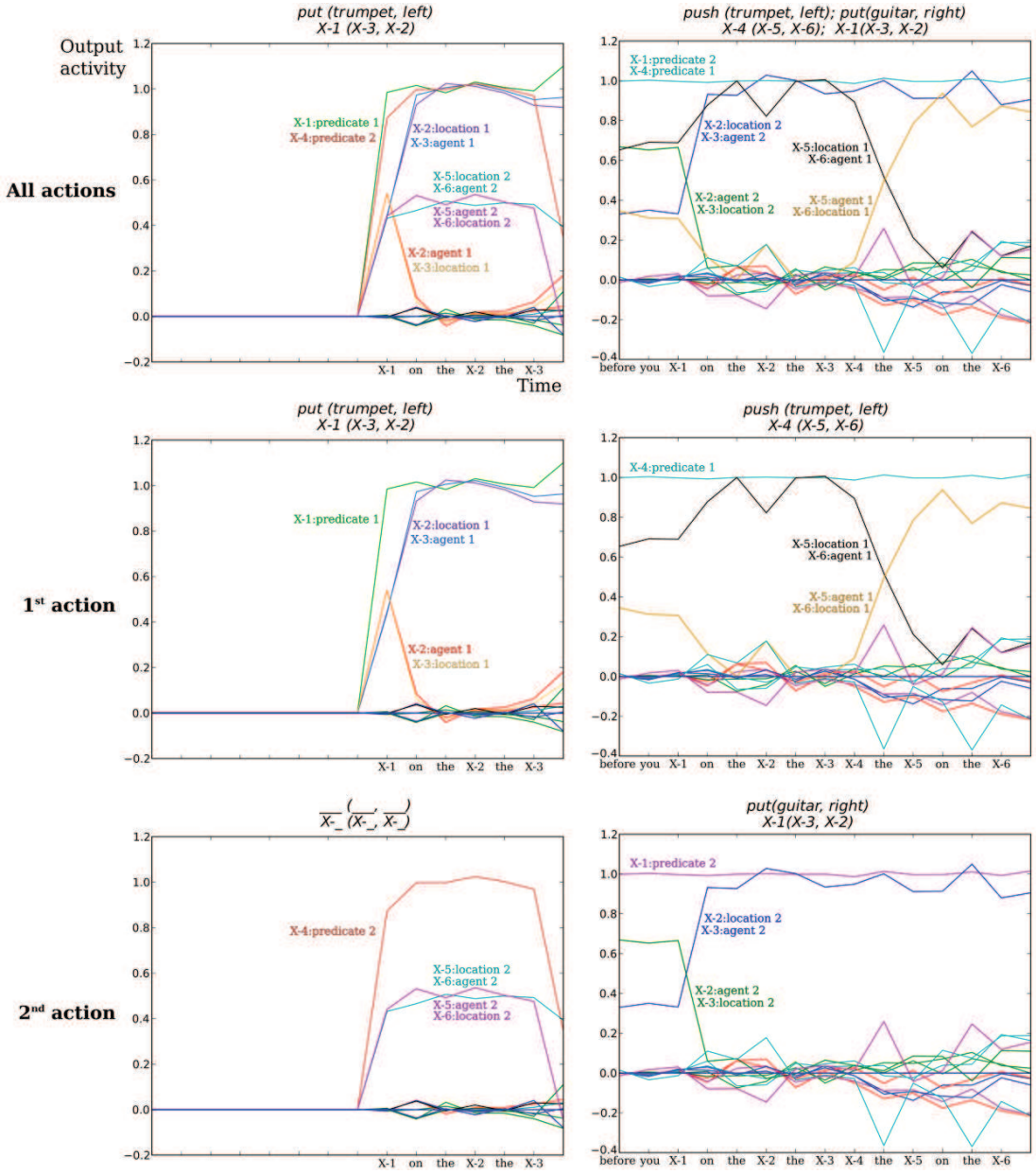


Figure 8: Example of output activity of the comprehension neural model for the "Action Performing" task. Each colored line indicates the pseudo-probability for each semantic word to have a given role (*predicate*, *agent*, *location*) for each of the two specified actions. (top) Output activity for both actions. (middle) Output activity for the first action to perform. (bottom) Output activity for the second action to perform. (left) The input sentence was "put on the left the trumpet". The model correctly determines that there is only one meaning-predicate *put (trumpet, left)*. X-1, X-2, X-3 … indicate the 1st , 2nd, 3rd, … SW markers. For X-5 and X-6 plots are superimposed, as the output neurons "X-5:location2" and "X-6:agent2" have the same activity for this sentence. (right) The input sentence was "before you put on the right the guitar push the trumpet on the left": the model correctly determines the two meanings in the right order *push (trumpet, left)* and then *put (guitar, right)*. Several curves are also superimposed.

For the Action Performer task, we show the activity for sentences that were not learned (i.e. not seen beforehand). Constructions shown in Figure 8 where not in the training data, but only in the test data. Even though the constructions were not pre-learned, the model was still able to correctly recognize them, demonstrating generalization capabilities. For more information on the model generalization performances see (Hinaut & Dominey 2012, Hinaut & Dominey 2013)

4.4.2 Production model neural activity for Scene Description task

Figure 9 illustrates the readout unit activations for two different meanings and different sentence forms in the Scene Description task. In Figure 9A, the meaning given in input was *right (trumpet, guitar)* with the sentence form *SW1(SW3, SW2)*. The model correctly generated the sentence "to the right of the guitar is the trumpet". In Figure 9B, the meaning given in input was {*right (violin, trumpet)*, *left (violin, guitar)*} with the sentence form {*SW1(SW5, SW2)*, *SW3(SW5, SW4)*}. The model correctly generated the sentence "to the right of the trumpet and to the left of the guitar is the violin".

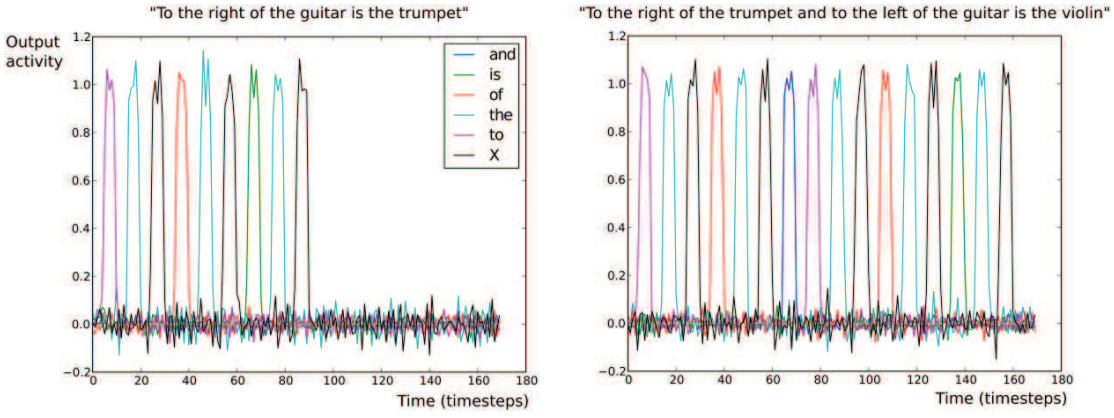


Figure 9: Output (read-out) unit activations of the production neural model in the Scene Description task. Each colored line represents a different read-out neuron. Each read-out neuron corresponds to a different word: either

a grammatical word or a SW marker. On the x-axis is indicated the number of time steps. On the y-axis is indicated the neural activity for output neurons. X indicates the semantic word (SW) marker. (left) The construction found is "To the X of the X is the X". The sentence correctly recovered after replacement of the SW markers is "To the right of the guitar is the trumpet". (right) The construction found is "To the X of the X and to the X of the X is the X". The sentence correctly recovered after replacement of the SW markers is "To the right of the trumpet and to the left of the guitar is the violin".

These results indicate that the system works correctly in the SD and AP tasks, under controlled conditions. We should also evaluate the capacity of the system to accommodate less controlled conditions. In Hinaut & Dominey (2013) we addressed the generalization capabilities of the sentence comprehension model with large corpora (up to 90K sentence-meaning pairs). In the current research we demonstrate that the model can learn and reuse grammatical constructions for sentence production. The extensive investigation of generalization properties (including the analysis of "incorrect" generated sentences) is beyond the scope of the current paper, and will be the subject of future research.

4.5 Action Performing Training with Naïve Subjects

Here we report on the results of the Action Performer model, when trained and tested with a set of sentences from five naïve subjects. Examination revealed that several additional closed class words were used by our subjects. They were used to define the set of possible inputs to the model. Here we defined the list of closed class word in a simple and systematic way: all the words that were not in the meaning part of the <sentence, meaning> pairs (i.e. the open class words that had a thematic role to be find) were defined as closed class words. Some of these words may appear once or a few times in the corpus, thus it is difficult for the model to learn their function. Please refer to Supplementary Material SM3 for the extended list of all 86 closed class words.

4.5.1 Naïve Subject Corpora

From the initial corpus of 380 sentences, a new corpus, where 7 <sentences, meaning> pairs were eliminated, was created : we will call the latter the 373 corpus. These 7 sentences did not fulfill the minimal conditions in order to be processed correctly by the system: they were ill-formed. For instance (1) they did not describe the actions properly (e.g. "make a U-turn": invention of new actions instead of using the atomic actions proposed), or (2) they did

not refer to the objects by their name (e.g. “touch both objects”). As we will see in the learnability analysis, these 7 sentences were part of the sentences that were not learnable by the system (see learnability test). The following analyses were performed on both initial and 373 corpora (see supplementary material SM4 to see all the <sentence, meaning> pairs of both corpora).

4.5.2 Learnability test

The analysis of the naïve subject data proceeded in two steps: learnability and generalization tests. We first tested the learnability capability of the complete set of sentences for each corpus: the reservoir was trained and then tested on the same set of sentences. Because of the increase in size and complexity of the training corpus compared to experiment 1 – subjects were ask to provide complex sentences structures –, we increased the reservoir size from 100 to 500 and 1000 neurons (for the generalization test). For the learnability test specifically, we deliberately took a large number of neurons (3000) in order to be sure that the system could learn the maximum <sentence, meaning> associations possible. Sentences are considered learnable if they were correctly learned “by heart” (i.e. without generalization) by the system. The learnability test results are taken as a reference for the generalization tests. The learnability test is based on the hypothesis that if the system is not able to learn some <sentence, meaning> associations by heart, then the system would not be able to generalize to such sentences. Thus the error obtained in the learnability test should be the lowest possible.

For the learnability test we created 4 instances of the model (i.e. different random generator seeds were used to generate the weight connections), but there is no variability between the results of these instances - learnable sentences are the same. Results for this learnability capability are illustrated in the “Learnability” column in Table 5. Only 16 sentences of the entire initial 380 corpus (i.e. 4.21%) were considered not learnable. Thus the vast majority of utterances produced by the naïve users were exploitable and learnable. This confirms the viability of the approach. For the 373 corpus, learnability error falls to 2.41% with only 9 sentences that are not learnable. These few sentences are not learnable because there is a competition between them that “interferes” because of an existing ambiguity among them. For instance for a same sentence structure (construction), some associations defines a meaning with 2 arguments, and others a meaning with 3 arguments: this lead to an ambiguity. Because some sentences are more frequent than others in the corpus, the latter could not be learnt: by definition the model selects the most probable solution for a given structure in case

of ambiguity. Ambiguity can for example be provoked by the use of irrelevant information in the sentence: "point the circle on my left" has the meaning *point(circle)* so "left" is irrelevant in this sentence, but the sentence "put the cross on my left" has the meaning *put(cross, left)* so left is important in this one; as the first type of structure (see sentences 152, 190 and 192 in SM4.1) has been used more frequently by the users than the second type (155, 156), the most frequent "desirable" behavior of the system is to ignore the open class word coming after "on my".

	Learnability Error		**Generalization Error (best)**	
Global	16/380 (4.2%)		133/380 (35.0%)	
	Single Action	Double Action	Single Action	Double Action
Simple Sentence	4/25 (16.0%)	9/165 (5.5%)	2/25 (8.0%)	44/165 (26.7%)
Elaborate sentence	0/25 (0.0%)	3/165 (1.8%)	9/25 (36.0%)	78/165 (47.3%)

Table 5. Learnability and best generalization capabilities on the naïve subject initial 380 corpus. (left) Learnability test performed with a reservoir of 3000 neurons. Number of non-learnable sentences for different sentence categories. For each category, the number of non-learnable sentences is divided by the total number of sentences for that category, with the corresponding percentage in parentheses. Only 4.2% of sentences are not learnable: this indicates that most of the corpus is learnable. (right) Best generalization errors for different sentence categories. For each category the neural model is able to generalize to some not learned sentences. As one could expect, generalization performances are better for Simple sentences than for Elaborate sentences. These results were obtained for a model of 1000 neurons using LoO method. No variability is observed when using such a number of neurons: the sentences that fail in generalization are always the same.

This learnability test is important to demonstrate the difficulty of the task, and it constitutes a preliminary step before looking at the generalization capability; because sentences that are not learnable have a priori no chance for being part of the group of sentences that the neural system could "understand" (i.e. generalize on). Of course the learnability of a sentence is also dependent on other sentences in the corpus: in this view, if one sentence is not learnable, it means that it is an outlier in this corpus.

4.5.3 Generalization test

In a second step we tested the ability of the model to generalize to sentences not used in training. We used a standard "leaving one out" (LoO) method: the model is trained on nearly all sentences and then tested on the sentence left out of the training data. This corresponds to the case were the robot-neural system has been taught hundreds of sentences and we want to test its ability to understand correctly a new sentence given by a naïve user. Even if that new sentence has a grammatical structure different from those in the training set, the system could nevertheless generalize to this untrained structure in some cases; this was demonstrated in (Hinaut & Dominey 2013). For this study, we used two sizes of reservoirs: 500 and 1000 neurons. We run 10 instances of the model for each size and each corpus.

For a reservoir size equal to 1000 units using the initial 380 corpus, 133 sentences failed to pass the generalization (LoO) test in all 10 simulations (i.e. for all 10 instances). We can consider that, for this amount of units in the reservoir (1000) – related to the computational power of the system – the corpus did not enable the system to have sufficient grammatical information to allow generalization to these 133 sentences. In Table 5, best generalization errors for different sentence categories are provided in the left column. The best generalization error over all categories is 35.0%. As expected, generalization error increase from Single to Double action sentences, and from Simple to Elaborate sentences. These results were obtained for a model of 1000 neurons also using LoO method. Considering the learnability results, which could lead to only 8.0% error for Simple – Single Action category, the system displays a good ability to generalize to unseen sentences. In particular, for the simple sentences (both single and double actions) the system is able to generalize to more than 75% of unseen sentences: this is an important result as in a natural conditions subjects will tend to produced spontaneously this type of sentences (that we categorized as "Simple").

4.5.4 Discussion on the "utility" of the learnability test

In Table 5, one could remark that for the Simple Sentence - Single Action category a lower error is obtained for the best generalization than for the learnability. This could be explained by the fact that LoO results with 1000N are the "best" accumulated over 10 simulations, thus it is possible that sometimes a given sentence that could not be learnt by heart with a reservoir of 3000 units (when nearly the whole corpus is correctly learned), could be generalized when using a smaller reservoir – here 1000 units – (when only a part of the corpus is correctly learnt). In particular if there is a "competition" between some sentences in the corpus that lead to an ambiguity. Consider that a group of sentences with a given construction A could not be learnt simultaneously with sentences with construction B; if more sentences of group A

than sentences of group B are learnt correctly, then the system could not learn (or generalize) correctly to sentences of the concurrent group B, and vice versa.

Indeed, this partly contradicts the hypothesis that was at the origin of the learnability test, because of sentences that could not be learnt with this test. However the "best" generalization results are not obtained with a single reservoir, but with 10 reservoirs running in parallel using the best possible combination of results of each reservoir – such an optimal combination may not be found without knowing in advance which reservoirs give the best answer for each sentence. Consequently, this is a demonstration that the learnability and generalization of a sentence is dependent on the corpus it constitutes, as the learning system tends to learn the corpus coherently. Thus outlier constructions that have poor chance to be learnt, which is a useful property if possibly ungrammatical constructions are present in the corpus. Here a part of ungrammaticality could also be interpreted as "less frequent", because for a learning system what makes a construction learnable (i.e. grammatical) is the fact that it has a higher probability of occurrence.

4.5.5 Summary of results for the generalization test

		Initial corpus	373 corpus
500 N	**Mean (std.)**	70.13 (1.87)	68.96 (2.03)
	Best	46,05	44.50
1000 N	**Mean (std.)**	58.53 (2.23)	58.26 (1.37)
	Best	35.00	34.85

Table 6. Generalization errors. Results for different conditions are shown: initial and 373 corpus and reservoir sizes of 500 and 1000 neurons. For each condition, the average error (Mean) over 10 instances along with the standard deviation (std.), and the best error (counts of only the errors in common within the 10 instances) are indicated. One can see that when the number of neuron increases, the negative influence of ill-formed sentences, removed in the 373 corpus, tend to decrease.

In Table 6. can be seen a summary of generalization errors for different conditions. A bigger reservoir (1000 compared to 500 neurons) clearly demonstrates better performances. On the contrary, the corpus does not have much influence on the performances. One can see that when the number of neuron increases, the negative influence of ill-formed sentences, removed in the 373 corpus, tend to decrease. Looking at the best error values, obtained when counting only errors that are made in common by all 10 instances, there is a clear decrease compared to the mean values. This big difference between best and mean values shows that

there is a high variability regarding which sentences are recognized by the different reservoir instances. This indicates that there is a clear potential to increase the performance of the system by combining several reservoirs in parallel. In addition, even better performance could be found by increasing the number of units in the reservoir, but this is not the point of the current study. We did not explore for the best parameters of the reservoir, we considered that the parameters we found in (X. Hinaut & Dominey, 2013) were sufficiently robust to be applied to a new type of corpus – produced by naïve users, demonstrating in this manner the robustness of this reservoir language model.

4.5.6 Some remarkable properties of the flexibility of the system

Some of the sentences that produced successful generalization are worth noting. Sentences (230), (245), (260) and (268) (see Table 7) illustrate the use of the impersonal pronoun "it" in various configurations of distinct constructions. Processed as a closed-class (i.e. grammatical) word, "it" indicates the appropriate role for the referent open class (i.e. semantic) element: the system is able to generalize correctly the function of the grammatical word "it" and bind to the correct role the semantic word it refers to. In a sentence like (230) (see Table 7) the second semantic word "circle" will be considered as the "object" of both actions, "grasp" and "point". Sentence (92) illustrates a similar situation, where the closed class word "twice" informs the system that the same action is repeated two times. Thus, in a certain sense, the system has learned the non-trivial meaning of the word "twice". Similarly, in sentence (313) this special function can be learned even when relying on several words: "two times". The system also acquires non-trivial use of the temporal relatives "before" and "after". In (198), (214) and (340), "before" is used in such a way that the first action appearing in the sentence is actually to be performed second. Thus in these situations, the presence of "before" results in a temporal inversion of the commanded actions. Interestingly, the system can also master a different use of "before" as illustrated in (5): here "before" does not result in an inversion, the order of actions in the sentence is preserved in the execution of actions. Similarly in sentence (268), "after" plays also the role of temporal inversion. Moreover such sentence illustrate how these different properties – "it": reference, "after": inversion – can be combined. Sentence (340) has a particular structure: "before" is the unique closed class word present in the sentence, the four open class words follow in a row. The system is nevertheless able to learn correctly this structure even if it could not distinguish the different open class words from one to another, because it does not have access to the semantics of these words. Sentences (198) and (214) have also the particularity to have

useless closed class words – for the given task – "please" and "you" have no specific function, but the model still has to learn to ignore these words. Although the system has not been designed to reach this level of "interpretation" of closed class words, it is able to generalize its use in not learned sentences. This ability of the system to work with non-predefined cases demonstrates its flexibility.

(5) point the triangle **before** grasping the circle
(20) put the cross to the left **before** grasping the circle
(92) point to the cross **twice**
(198) **before you** grasp the cross **please** grasp the triangle
(214) **before** pushing the triangle to the middle **please** push the cross to the right
(230) grasp the circle and then point to **it**
(245) touch the triangle then move **it** to the left
(260) the cross touch **it**
(268) point to the circle **after** having grasped **it**
(313) point cross **two times**
(340) **before** grasp circle point triangle

Table 7. Example sentences produced by naïve subjects (of the 373 corpus; see SM4.2), and understood by the model (i.e. 0% error in LoO generalization simulations for a reservoir of 1000 units). Closed class words indicated in bold have a specific function and the system has to learn it without any additional feature to treat these special words. These words are common words of natural language, but they are not essential to form a correct sentence understandable by the system. Nevertheless, the system is able to learn their specific function. Numbers in parenthesis indicate the identifiers of the sentences for the corpus 373; note that identifiers are not the same for initial and 373 corpora.

5. Discussion

The current research makes several distinct contributions to language-based human-robot interaction. Previous research has used language to command humanoids e.g. (Dominey et al 2007, Lallée et al 2012, Petit et al 2013), and to allow robotic systems to describe actions (Dominey & Boucher 2005b). The current research for the first time demonstrates real-time acquisition of new grammatical constructions for comprehension and production that can be used respectively in commanding the robot and in asking the robot to describe the physical world. This is of interest both in theory and in practice. In theory, it demonstrates that the form-to-meaning mapping that we have employed in learning grammatical constructions can be used in both directions, to generate meaning from form, and to generate form from

meaning. In practice, this means that the system can adapt to individual differences in the way users employ language to describe and request actions. The current research also addresses how language can allow for the coordination of multiple sub-actions in time, using the prepositions "before," and "after." Learning of these terms has a long history of research in child language development, and it poses an interesting problem because of the interaction with non-temporal event ordering and non-canonical syntactic structure (Carni & French 1984). Our work can contribute to the debate by indicating that a system that is sufficiently powerful to handle canonical and non-canonical events in single and double event sentences can do the same in sentences in which order is expressed with prepositions including "before" and "after". Interestingly, the key assumption is that these prepositions are processed in the model as closed class or grammatical words, which can then directly contribute to the elaboration of the form to meaning mapping.

Because of this flexibility, the framework that we have developed potentially enables naive users to interact with the robot, indeed there is no "predefined" way of giving a command or description of an action such as *put (toy, left)* ; the user could say "put the toy on the left" or "on the left put the toy". In this way, we are able to escape from a 1-to-1 sentence-action correspondence: several sentences could indicate the same meaning.

Concerning the production model we partly escape the 1-to-1 sentence-action (or sentence-scene) limitation because we can specify if we want a canonical or non-canonical sentence type. We could specify a more precise sentence type, for instance by specifying the semantic word of focus. But this problem could be tackled in a more general way. In order to be able to generate several sentences with the same meaning, we could consider 2 alternatives. (1) We could add feedback connections from the readout layer to the reservoir with the addition of noise either in the reservoir states or in these feedback connections. Thus the network would not produce every time the same pattern of words, but different ones. The noise would enable the network to be driven by one of the possible learned sentences (word patterns). (2) Use an additional self-organization map (SOM) based on the semantic words. During training this SOM will tend to organize words that appear in the same sentences in the same area of the map. During testing, the SOM activation will provide a supplementary input to the sentence production model in order to give a kind of context and enable the model to generate one pattern of words that is context relevant. In this way, if some sentence constructions are commonly used with certain semantic words, it will produce the more common sentences. Both alternative solutions may enable the production of constructions that were not learned, i.e. give the production generalization capabilities (like the comprehension model). Finally,

the generation of different non-canonical forms allows the system to manipulate the grammatical focus while describing the same situation, as illustrated in Table 1.

The production model introduced here is able to learn to produce grammatical constructions when given the meaning, coded in the same way that the comprehension model output is coded. This is the first time that we demonstrate that the input and output of the comprehension model could be reversed in order to do the "inverse" task (i.e. production instead of comprehension). This is an interesting property that may be useful in further understanding human language. Indeed, we have here a system that is able to do grammatical construction comprehension and production with a common coded meaning representation (which corresponds to the output of the comprehension model, and to the input of the production model). We can imagine that the two models can be running in parallel, with the outputs of the production model connected to the inputs of the comprehension model. In this way, when the production model would be generating a sentence, the latter could be decoded in real-time and fed to the inputs of the comprehension model. Thus the comprehension model will reconstruct in real-time the meaning of the sentence produced by the production model. Consequently this would allow the system to check if the produced sentence is correct or not to the original meaning (i.e. the input of the production model). A correction mechanism could be then added to compensate when errors of productions are made. Such a correction mechanism appears to exist in human language behavior, as when one notices that they have produced a word instead of another in the middle of a sentence, they correct their sentence production in real-time accordingly. Detection of such a production error would likely be accompanied by specific brain response, as it is the case for the P600 event related scalp potential when an ungrammatical word or complex sentence is processed. In a previous study using our comprehension model (Hinaut & Dominey 2013) we showed that a kind of instantaneous derivative of the output values – the sum of absolute change of all outputs – could be related with a P600-like event. In the reverse sense, the output of the comprehension model could be input to the production model, allowing the listener to predict the upcoming words of the speaker. Another alternative would be to combine both comprehension on production within a same model, with feedback connections from both discovered thematic roles and produced words: a unique reservoir would do both tasks at the same time; this would probably require an online learning algorithm.

The experiment with the naïve subjects is particularly interesting, as it provides the model with a form of "cognitive variability" in the language used, which goes beyond that employed when "insider" researchers interact with the robot. The use of the impersonal pronoun "it",

words like “twice”, the use of “before,” and “after,” in the diverse configurations allowed a test and finally an illustration of the adaptability of the language model. The good learnability of the sentences – 93% of the corpus is learnable – indicates that the naïve subjects can make really complicated sentences that may contain only partial information. The relatively robust generalization, particularly for the “simple” sentences (>75% generalization) indicates that the model was able to extract the relevant information from this relatively small (< 400 sentences) corpus; it also indicates that the naïve subjects are “playing the game,” i.e. they are attempting to speak in a reasonable way to the robot in the “simple” sentence condition. Future research should asses how, as such corpora increase in size, generalization improves (for a given corpus complexity), as indicated in (Hinaut & Dominey 2013).

6. Acknowledgments

This work has been financed by the FP7-ICT 231267 Project Organic and by the FP7-ICT-270490 Project EFAA. Neural model has been developed with Oger toolbox: http://reservoir-computing.org/organic/engine.

7. Video links

Video demonstration of the scene description in Experiment 1 can be seen at:
http://youtu.be/AUbJAupkU4M
Video demonstration of the action performer in Experiment 2 can be seen at:
http://youtu.be/3ZePCuvygi0

Supplementary Material

SM1: Input file for Action Performer task with train and test data used for Figure 8.

SM2: Input file for Scene Description task with train and test data used for Figure 9.

SM3: Details concerning the Naïve Subject Experiment : List of closed class words used and removed sentences from raw corpus for the naïve subject experiment; List of the 7 <meaning, sentence> pairs removed from the raw corpus of 380 sentences (i.e. sentences that are not present in 373 corpus).

SM4: Files containing the detailed results of the naïve subject experiment for the learnability test for both corpora (initial or 373 corpus) with a reservoir size of 3000 units.

(See data sheet files 4, 5 and 6.)

SM5: Files containing the detailed results of the naïve subject experiment for the generalization test for the different conditions: initial or 373 corpus, and 500 or 1000 reservoir units.

(See data sheet files 7, 8, 9 and 10.)

SM6: Movie scenes shown to the users for the naïve subject experiment.

References

Bates E, MacWhinney B. 1987. Competition, variation, and language learning In *Mechanisms of language acquisition*, ed. B MacWhinney, E Bates, pp. 157-93. Hillsdale, NJ: Erlbaum

Bates E, McNew S, MacWhinney B, Devescovi A, Smith S. 1982. Functional constraints on sentence processing: a cross-linguistic study. *Cognition* 11: 245-99

Bencina R, Kaltenbrunner M, Jorda S. *Computer Vision and Pattern Recognition-Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on2005*: 99-99. IEEE.

Blanc JM, Dodane C, Dominey PF. *The 25th Ann Conf. Cog. Sci. Soc.,, Cambridge, MA,, 2003*.

Carni E, French LA. 1984. The acquisition of 'before' and 'after' reconsidered: What develops? *Journal of experimental child psychology* 37: 394-403

Carpenter M, Nagell K, Tomasello M. 1998. Social cognition, joint attention, and communicative competence from 9 to 15 months of age. *Monogr Soc Res Child Dev* 63: i-vi, 1-143

Dominey P, Boucher J. 2005a. Developmental stages of perception and language acquisition in a perceptually grounded robot. *Cognitive Systems Research* 6: 243-59

Dominey P, Boucher J. 2005b. Learning to talk about events from narrated video in a construction grammar framework. *Artificial Intelligence* 167: 31-61

Dominey P, Mallet A, Yoshida E. *IEEE International Conference on Robotics and Automation2007*: 2169-74.

Dominey PF. 1995. Complex sensory-motor sequence learning based on recurrent state representation and reinforcement learning. *Biological cybernetics* 73: 265-74

Dominey PF. *25th Annual Meeting of the Cognitive Science Society, Boston, 2003*.

Dominey PF, Arbib MA, Joseph JP. 1995. A Model of Corticostriatal Plasticity for Learning Oculomotor Associations and Sequences *J Cogn Neurosci* 7: 25

Dominey PF, Dodane C. 2004. Indeterminacy in language acquisition: the role of child directed speech and joint attention. *Journal of Neurolinguistics* 17: 121-45

Dominey PF, Hoen M, Blanc JM, Lelekov-Boissard T. 2003. Neurological basis of language and sequential cognition: evidence from simulation, aphasia, and ERP studies. *Brain Lang* 86: 207-25

Goldberg A. 1995. *Constructions: A Construction Grammar Approach to Argument Structure*. Chicago: University of Chicago Press. 265 pp.

Goldberg AE. 2003. Constructions: a new theoretical approach to language. *Trends Cogn Sci* 7: 219-24

Gori I, Pattacini U, Nori F, Metta G, Sandini G. *IEEE/RSJ International Conference on Intelligent Robots and Systems2012*.

Hinaut X, Dominey P. 2012. On-Line Processing of Grammatical Structure Using Reservoir Computing In *ICANN 2012 - Lecture Notes in Computer Science*, ed. A Villa, W Duch, P Érdi, F Masulli, G Palm, pp. 596-603: Springer Berlin / Heidelberg

Hinaut X, Dominey PF. 2011. A three-layered model of primate prefrontal cortex encodes identity and abstract categorical structure of behavioral sequences. *Journal of physiology, Paris* 105: 16-24

Hinaut X, Dominey PF. 2013. Real-time parallel processing of grammatical structure in the fronto-striatal system: a recurrent network simulation study using reservoir computing. *PloS one* 8: 1-18

Huang X, Alleva F, Hon H-W, Hwang M-Y, Lee K-F, Rosenfeld R. 1993. The SPHINX-II speech recognition system: an overview. *Computer Speech & Language* 7: 137-48

Jaeger H. 2001. The" echo state" approach to analysing and training recurrent neural networks-with an erratum note'. *Bonn, Germany: German National Research Center for Information Technology GMD Technical Report* 148

Jaeger H, Haas H. 2004. Harnessing nonlinearity: predicting chaotic systems and saving energy in wireless communication. *Science* 304: 78-80

Knoblich G, Sebanz N. 2008. Evolving intentions for social interaction: from entrainment to joint action. *Philos Trans R Soc Lond B Biol Sci* 363: 2021-31

Lallée S, Pattacini U, Lemaignan S, Lenz A, Melhuish C, et al. 2012. Towards a Platform-Independent Cooperative Human-Robot Interaction System: III. An Architecture for Learning and Executing Actions and Shared Plans. *IEEE Transactions on Autonomous Mental Development* 4: 239-53

Lukosevicius M, Jaeger H. 2009. Reservoir computing approaches to recurrent neural network training. *Computer Science Review* 3: 22

Maass W, Natschlager T, Markram H. 2002. Real-time computing without stable states: a new framework for neural computation based on perturbations. *Neural Comput* 14: 2531-60

Metta G, Fitzpatrick P, Natale L. 2006. YARP: yet another robot platform. *International Journal on Advanced Robotics Systems* 3: 43-48

Metta G, Natale L, Nori F, Sandini G, Vernon D, et al. 2010. The iCub Humanoid Robot: An Open-Systems Platform for Research in Cognitive Development. *Neural Networks, Special issue on Social Cognition: From Babies to Robots* 23

Morgan JL, Demuth K. 1996. *Signal to syntax: boostrapping from speech to grammar in early acquisition*. Mahwah, NJ: Lawrence Erlbaum.

Petit M, Lallee S, Boucher J-D, Pointeau G, Cheminade P, et al. 2013. The Coordinating Role of Language in Real-Time Multi-Modal Learning of Cooperative Tasks. *IEEE Transactions on Autonomous Mental Development* 5: 3-17

Piattelli-Palmarini M. 1980. *Language and learning: the debate between Jean Piaget and Noam Chomsky*. Cambridge Univ Press.

Ricciardelli P, Bricolo E, Aglioti SM, Chelazzi L. 2002. My eyes want to look where your eyes are looking: exploring the tendency to imitate another individual's gaze. *Neuroreport* 13: 2259-64

Sebanz N, Bekkering H, Knoblich G. 2006. Joint action: bodies and minds moving together. *Trends Cogn Sci* 10: 70-6

Siskind JM. 1996. A computational study of cross-situational techniques for learning word-to-meaning mappings. *Cognition* 61: 39-91

Sutton S, Cole RA, De Villiers J, Schalkwyk J, Vermeulen PJ, et al. *ICSLP1998,* 98: 3221-24.

Taylor P, Black AW, Caley R. 1998. The architecture of the Festival speech synthesis system. In *Proc. Third ESCA/COCOSDA Workshop on Speech Synthesis, Jenolan Caves House, Blue Mountains, Australia, November 26-29, 1998. pp. 147-151.*

Tomasello M. 2003. *Constructing a language: A usage based approach to language acquisition*. Boston: MIT Press.

Tomasello M, Hamann K. 2012. Collaboration in young children. *Q J Exp Psychol (Hove)* 65: 1-12

Verstraeten D, Schrauwen B, D'Haene M, Stroobandt D. 2007. An experimental unification of reservoir computing methods. *Neural Networks* 20: 391-403

Chapter 4

The Coordinating Role of Language in Real-Time Multimodal Learning of Cooperative Tasks

4.1 Introduction

Now that the iCub has some linguistic capabilities, we have worked on what it could bring to his development. Yet, one on the main function of language is to coordinate between the agents during cooperative activity ([Brinck and Gärdenfors, 2003, Tomasello, 2008]). Based on joint attentional skill, the children has to see other as intentional agents like themself ([Tomasello et al., 2005]) in order to participate in a true collaborative task. By using demonstration or true imitation, they can learn new actions and produce a shared plan with others, using language to coordinate with their partner and negociate the role or sharing their intentions ([Carpenter et al., 2005, Tomasello et al., 2005]).

4.2 Publication

The Coordinating Role of Language in Real-Time Multimodal Learning of Cooperative Tasks

Maxime Petit, Stéphane Lallée, Jean-David Boucher, Grégoire Pointeau, Pierrick Cheminade, Dimitri Ognibene, Eris Chinellato, Ugo Pattacini, Ilaria Gori, Uriel Martinez-Hernandez, Hector Barron-Gonzalez, Martin Inderbitzin, Andre Luvizotto, Vicky Vouloutsi, Yannis Demiris, Giorgio Metta, and Peter Ford Dominey

***Abstract*—One of the defining characteristics of human cognition is our outstanding capacity to cooperate. A central requirement for cooperation is the ability to establish a "shared plan"—which defines the interlaced actions of the two cooperating agents—in real time, and even to negotiate this shared plan during its execution. In the current research we identify the requirements for cooperation, extending our earlier work in this area. These requirements include the ability to negotiate a shared plan using spoken language, to learn new component actions within that plan, based on visual observation and kinesthetic demonstration, and finally to coordinate all of these functions in real time. We present a cognitive system that implements these requirements, and demonstrate the system's ability to allow a Nao humanoid robot to learn a nontrivial cooperative task in real-time. We further provide a concrete demonstration of how the real-time learning capability can be easily deployed on a different platform, in this case the iCub humanoid. The results are considered in the context of how the development of language in the human infant provides a powerful lever in the development of cooperative plans from lower-level sensorimotor capabilities.**

***Index Terms*—Cooperation, humanoid robot, shared plans, situated and social learning, spoken language interaction.**

I. Introduction

THE ability to cooperate, creatively establish, and use shared action plans is, like language and the underlying social cognitive and motivational infrastructure of communication, one of the major cognitive capacities that separates humans from nonhuman primates [1]. In this context, language itself is an inherently cooperative activity in which the listener and speaker cooperate, in order to arrive at the shared goal of communication. Tomasello *et al.* make the foundational statement that language is built on the uniquely human ability to read and share intentions, which is also the foundation for the uniquely human ability and motivation to cooperate. Indeed, Tomasello goes one step further, suggesting that the principal function of language is to establish and negotiate cooperative plans [1].

The building blocks of cooperative plans are actions. In this context, it has been suggested that we are born with certain systems of "core cognition," which are "identified by modular innate perceptual-input devices" [2]. One of the proposed elements of core cognition is agency. This includes an innate system for representing others in terms of their goal directed actions, and perceptual mechanisms such as gaze following that allow the developing child to monitor the goal directed actions of others. Thus we consider that these notions of agency are given in the system, though the degree to which they may actually be developed versus innate remains an open question [2].

A cooperative plan (or shared plan) is defined as a goal directed action plan, consisting of interlaced turn-taking actions by two cooperating agents, in order to achieve a common goal that could otherwise not have been achieved individually [1]. Interestingly, infants can establish shared plans without the use of language, if the shared goal and corresponding plan are sufficiently simple. However, once the plans reach a certain level of complexity, and particularly if the plan must be renegotiated in real-time, then language is often invoked to establish and negotiate who does what [3], [4]. Thus, cooperation requires communication, and when things get complex, language is the preferred communication method. Indeed, much of early language maps onto physical parameters of goal directed action [5], [6].

In the construction grammar framework, Goldberg identifies how the structure of language is mapped onto the structure of meaning such that "constructions involving basic argument structure are shown to be associated with dynamic scenes . . . such as that of someone volitionally transferring something to someone else, someone causing something to move or change state" [5]. Thus, grammatical constructions implement the mapping from linguistic utterances to meaning, in the form of action and perceptual scene specifications. The nature of the link between language and action, and how that link is established, is an open topic of research in child development and developmental robotics [7].

Manuscript received January 04, 2012; revised May 03, 2012 and June 15, 2012; accepted June 25, 2012. Date of publication July 26, 2012; date of current version March 11, 2013. This work was supported by the European Commission under Grants EFAA (ICT-270490) and CHRIS (ICT-215805).

M. Petit, S. Lallee, J.-D. Boucher, G. Pointeau, P. Cheminade, and P. F. Dominey are with SBRI, Institut National de la Santé et de la Recherche Médicale (INSERM), Bron 69675, France (e-mail: maxime.petit@inserm.fr; stephane.lallee@inserm.fr; jean-david.boucher@inserm.fr; greg.pointeau@gmail.com; kaosumog@gmail.com; peter.dominey@inserm.fr).

D. Ognibene, E. Chinellato, and Y. Demiris are with the Department of Electrical and Electronic Engineering, Imperial College London, London SW7 2BT, U.K. (e-mail: d.ognibene@imperial.ac.uk; e.chinellato@imperial.ac.uk; y.demiris@imperial.ac.uk).

U. Pattacini, I. Gori, and G. Metta are with the Cognitive Robotics Department, Italian Institute of Technology, Genoa 16163, Italy (e-mail: ugo.pattacini@iit.it; ilaria.gori@iit.it; giorgio.metta@iit.it).

U. Martinez-Hernandez and H. Barron-Gonzalez are with the Department of Psychology, University of Sheffield, Sheffield, U.K. (e-mail: uriel.martinez@sheffield.ac.uk; hector.barron@sheffield.ac.uk).

M. Inderbitzin, A. Luvizotto, and V. Vouloutsi are with SPECS, Universitat Pompeu Fabra, Barcelona, Spain (e-mail: martin.inderbitzin@upf.edu; luvizotto@gmail.com; vidago@gmail.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TAMD.2012.2209880

In the context of this debate, following a usage-based approach [6], we have demonstrated how such constructions can be learned in a usage-based approach, as the mapping between the argument structure of sentences and argument structure of robotic representations of action meanings [8]. This "usage-based" development of grammatical constructions (versus a more nativist approach) is also a topic of debate, similar to the case for agency cited above.

Independent of the nativist vs. usage-based debate, we can take the position that via such constructions, language is uniquely situated in its capability to allow agents to construct and negotiate shared cooperative plans. Our approach is to implement a scaffolded system based on this capability. In this scaffolding, we build in simple grammatical constructions that map onto the argument structure of actions that can be performed by the robot. This allows a scaffolding for the creation of action plans. We have previously used spoken language to construct diverse action plans for a robot cooperating with a human [9], [10], but the plans were not shared, in that they only specified the robot's actions. We then introduced a shared planning capability where a robot could observe a sequence of actions, with an agent attributed to each by the user via language. This generated a true shared plan, that could pass the test of role reversal [11]. Role reversal occurs when the two participants in a cooperative task can exchange roles, thus indicating that they both have a "bird's eye view" of the shared plan, which is a central part of the requirements for true cooperation [12].

In a series of studies we then more carefully reexamined the bases of shared planning. In the first study [13], we implemented a capability for learning to perceive and recognize novel human actions based on the structure of perceptual primitive constituting those actions. We next implemented the corresponding ability to learn to execute complex actions based on the composition of motor primitives, and to make the link between perception and action via imitation [14]. Finally, we extended this capability to multiple actions in shared plans, where the human could use spoken language to specify a shared plan that could then be executed by the robot, again displaying role reversal [15].

While this work represented significant progress, it left several issues unanswered. First, when a shared plan "goes wrong" there is no mechanism to fix it. Language can fulfill this role-indeed much of human language is about coordinating and correcting shared plans [16]. Second, in our previous work, teaching the shared plan was in a fixed modality, typically with the human speaking the shared plan, action by action. Here we extend this so that language becomes the central coordinator, a scaffold, which allows the user to then specify individual actions by: 1) kinesthetically demonstrating the action; 2) performing the action himself so the robot can perceive and imitate; or 3) for known actions—to specify the action verbally. Learning by visual and kinesthetic demonstration are highly developed and well documented means for transmission of skill from human to robot, e.g., [17]–[19]. We will demonstrate how this provides a novel interaction framework that where language coordinates these three potential modalities for learning shared plans.

The transmission of knowledge from humans to robots can take multiple forms. We consider three specific forms. "Imitation" will refer to learning in which the human performs the action to be learned, and the robot observes this and performs a mapping from observation space onto its execution space, as defined in [20]. Likewise, based on [20] we will refer to "kinesthetic teaching" as a form of "demonstration" where the passive robot is moved through the desired trajectory by the human teacher. Finally we will refer to "spoken language programming" [21] as the method described above where well-formed sentences are used to specific robot actions and arguments, either in isolation or in structured sequences. Language has been used to explain new tasks to robots [22], and is especially useful for scaffolding tasks, when the teacher uses previously acquired skills to resolve a new and more complex tasks [23].

Imitation has been successfully used on diverse platforms [24]–[29]. It is an easy way for the teacher to give the robot the capacity to perform novel actions, and is efficient in high dimensional spaces, and as a mechanism for communication [30]. It also speeds up the learning time by reducing the repetitions required for trial-and-error learning [31], and it can lead to open-ended learning without previous knowledge of the tasks or the environment [32].

Demonstration (also called self-imitation) [33], [34] avoids the problem of mapping from teacher to observer space. While this problem exists during imitation, it is eliminated in demonstration, as the human directly move the limbs of the robot [20] thus avoiding the "Correspondance Problem" [28]. It also does not require expert-knowledge of the domain dynamics, allowing the teacher to be a nonexpert [20].

Some authors have also studied multimodal learning, combining these techniques; including imitation and instructions [35]–[37] or demonstration and instruction [38]. In this research we build upon and extend these multimodal approaches. We implement a multimodal learning architecture which allow a user to teach action to robots (iCub and Nao) using one or a combination of language instructions, demonstration or imitation. More precisely, demonstration is a form of "tele-operation" by "kinesthetic teaching" and imitation is mediated by "external sensor" as defined in [20]: demonstration by kinesthetic teaching because the teacher operates directly on the robot learner platform, and imitation by external sensor because we are using kinect as perceptual device to encode the executing body's moves.

Thus the novelty of the current research is threefold—first it demonstrates a rich language capability for establishing and negotiating shared plans in real time. Second, it does this by allowing a multimodal combination of spoken language programming, imitation and demonstration based learning. Finally, it demonstrates that, with an appropriate robotic platform, language can be used as the glue that binds together learning from these different modalities. These capabilities are demonstrated on two robots, the Nao and the iCub, which allow us to take advantage of the specific motor capabilities of each, including the more dexterous manipulation capabilities of the iCub.

II. System Requirements and Design

The goal of the current research is to demonstrate that a learning system that is based on the human developmental

capability to map language onto action can provide the basis for a multimodal shared plan learning capability. In order to proceed with this analysis, we consider a scenario that involves multimodal learning. This will allow us in particular to determine the requirements involved in a human–robot cooperation to achieve an unknown task with real-time learning.

Consider a scenario where a humanoid robot and a human are in a face-to-face interaction, with a box and a toy put on a table. The human wants to clean the table, by putting the toy in the box. In order to do that, he must first grasp the toy, then open the box, then put the toy in the box, and finally close the box. Let us further consider that the human cannot grasp the toy and open the box at the same time, and that he thus needs help in performing this task. The human will ask the robot to "clean the table." The robot doesn't yet know the plan so it will ask the human to explain. The user will describe each step of the plan, which is composed by several sequential actions:

— "I grasp the toy, then;
— you open the box, then;
— I put the toy in the box, then;
— you close the box."

After checking whether the stated shared plan has been understood correctly, the robot will check each action that it should perform. The robot recognizes that there are some problems because it does not know how to open or close the box. It will ask for the help of the human, who has to teach it however he wants.

For opening the box, the human will decompose the teaching in two parts: at first, going to a safe initial position and next imitating him. After the opening action is learned, the user will teach the closing behavior, by directly demonstrating the motion by moving the arm of the robot. Finally, the robot has learned the whole shared plan and each action it should perform, and so the two agents can proceed and clean the table together. This scenario allows us to identify the functional requirements for the system. The system should:

1) inderstand human language, including mapping grammatical structure onto internal representation of action;
2) appropriately distinguish the definition of self and the other for relative pronouns (e.g., "I," "You");
3) manage a memory of known shared plan and actions;
4) become active in the discussion by asking human when a problem occurred;
5) perform Inverse kinematics mapping to learn from human action by imitation;
6) encode proprioception induced when the human is moving the robot to teach;
7) perceive the state of objects in the world.

In the following sections, we will define an overall system architecture that accommodates requirements 1)–4) in a platform independent manner, suggesting that these are the core learning functions. We will further demonstrate how this system can be used for real-time multimodal shared plan learning on the Nao with requirements 5) and 6), and on the iCub with point 7).

III. System Design Overview

Here, we present the system architecture for the learning and execution of cooperative shared plans. We begin with the components that are independent of the physical platform, and then introduce the platform specific components.

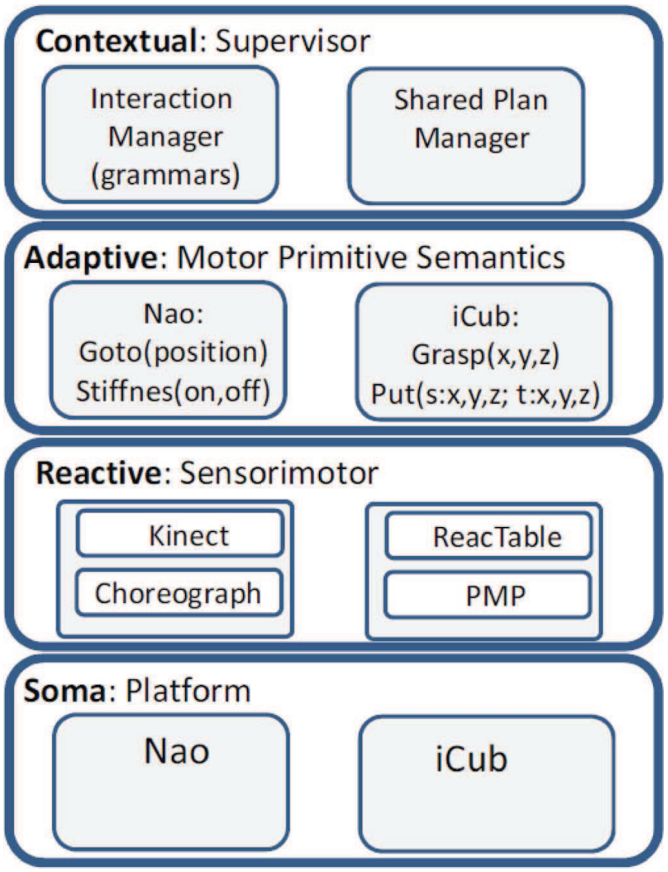


Fig. 1. Biomimetic Architecture for Situated Social Intelligence Systems (BASSIS).

The BASSIS architecture (see Fig. 1) is a multiscale architecture organized at three different levels of control—reactive, adaptive, and contextual, where the different levels of self are all based on the physical instantiation of the agent through its body (soma). It is based on the distributed adaptive control architecture [39]–[41]. Soma corresponds to the physical platform, instantiated as the Nao or iCub in our experiments. The Reactive or sensorimotor layer employs Kinect for perception and Choreograph™ (Aldebaran) for motor control on the Nao, and the ReacTable sensitive table, and the passive motion planner (PMP) and iKin inverse kinematic solver for iCub. The Adaptive layer defines adaptive motor capabilities for each robot. In the current context, this adaptation can take place through learning within the human–robot interaction. The Contextual layer is platform independent, and implements a Supervisor function, with a grammar-based Interaction Manager, and a Shared Plan Manager. Within the BASSIS framework, the Contextual layer implements a form of long term memory that we exploit here in the context of learning shared action plans.

A. Supervisor

The Supervisor function consists in two related capabilities. The first is general management of the human–robot interaction via a state-based dialog management capability. The second is

capability to learn and execute shared plans. Both of these functions are platform independent.

1) Interaction Management: Interaction management is provided by the CSLU Toolkit [42] rapid application development (RAD) state-based dialog system which combines state-of-the-art speech synthesis (*Festival*) and recognition (Sphinx-II recognizer) in a GUI programming environment. RAD allows scripting in the TCL language and permits easy and direct binding to the YARP domain, so that all access from the Interaction Management function with other modules in the architecture is via YARP.

The system is state-based with states for specifying the shared plan, modifying the shared plan, if there are errors, teaching specific actions within the shared plan, and finally, executing the shared plan during the cooperative task execution. Interaction management also allows the system to indicate error states to the user, and to allow him to explore alternate possibilities to rectify such errors, as illustrated in Section IV-A.2.

2) Shared Plan Learning: The core aspect of the learning capability is the capability to learn and execute shared plans, and to learn constituent actions that can make up those plans. As defined above, a shared plan is a sequence of actions with each action attributed to one of two agents in a turn-taking context. Shared plans can be learned via two complimentary learning mechanisms. The first method involves a form of spoken language programming, in which the user verbally describes the succession of action-agent components that make up the shared plan. Recognition is based on a grammar that we have developed for this purpose:

1) $SharedPlan = pedro%%*sil%% $agent $command [(($linkWord $agent $command))];
2) $agent = you | I;
3) $command =
   a. $action1 [*sil%%]|
   b. $action2 [*sil%%];
4) $pause = [*sil%%][*any%%][*sil%%];
5) $object = winnie | toy | chest;
6) $posture = initial−position;
7) $action1 =
   a. grasp $pause $object|
   b. reach $pause $object|
   c. open $pause $object|
   d. close $pause $object|
   e. move−to $pause $posture;
8) $action2 = put $pause $object $pause [in%%] $pause $object;
9) $linkWord = then | after−that | next | and;

Line (1) specifies that a shared plan begins with the "imperative" "Pedro" (the robot's name) followed by an optional silence (*sil%%), then an agent and command, followed by [0-n] groups made of a link word, an agent and a command. Agent terminals are identified in (2). Commands can take 1 or two arguments, as specified, respectively, in 7) and 8). Interestingly, in this grammar, the set of terminal nodes (actual words to be recognized) is only 16 distinct words. Thus, the speaker independent recognition system is in a well-defined recognition niche, and the system works with few to no errors.

In the case that errors are made, either in recognition, or by the user forgetting a command, saying a wrong command etc. we have a "spoken language programming" editing capability. Editing can involve the following edits: replace one command with another. In this case the user repeats the faulty command, and then the correct one (in cooperation with the dialog system of the robot). Delete a command, in which case the user stats the command to be deleted. Insert a command, in which case the user says before or after a given command, and then the new command.

The second learning mechanism is evoked at the level of individual actions, and allows the user to teach new component actions to the robot. This involves a combination of spoken language programming and perceptual action recognition. Perceptual action recognition can occur via action recognition with the Kinect, and via kinesthetic demonstration, which will be detailed below. The robot can then use the resulting shared plan to take the role of either agent, thus demonstrating the crucial role-reversal capability that is the signature of shared planning [1], [12].

As illustrated in the example dialog with the Nao below, this provides a rich capability to negotiate a complex cooperative task using spoken language. The resulting system can learn how to perform novel component actions (e.g., open, close), and most importantly, it can learn arbitrary novel turn-taking sequences—shared plans—that allow the user to teach in any novel cooperative behavior to the robot in real-time. The only constraint is on the set of composite actions from which the novel behavior can be constructed.

B. YARP

Software modules in the architecture are interconnected using YARP [43], an open source library written to support software development in robotics. In brief YARP provides an intercommunication layer that allows processes running on different machines to exchange data. Data travels through named connection points called ports. Communication is platform and transport independent: processes are not aware of the details of the underlying operating system or protocol and can be relocated at will across the available machines on the network. More importantly, since connections are established at runtime it is easy to dynamically modify how data travels across processes, add new modules or remove existing ones. Interface between modules is specified in terms of YARP ports (i.e., port names) and the type of data these ports receive or send (respectively for input or output ports). This *modular* approach allows minimizing the dependency between algorithm and the underlying hardware/robot; different hardware devices become interchangeable as long as they export the same interface.

C. Humanoid Robot Nao and Kinect

The Nao (Fig. 3) is a 25 degrees of freedom humanoid robot built by the French company Aldebaran. It is a medium size (57 cm) entertainment robot that includes an onboard computer and networking capabilities at its core. Its open, programmable and evolving platform can handle multiple applications. The onboard processor can run the YARP server (described below) and can be accessed via telnet connection over the internet via WiFi.

More specifically, the Nao is equipped with the following: CPU x86 AMD Geode with 500 MHz, 256 MB SDRAM

and 1 Gb Flash memories, WiFi (802.11g) and Ethernet, $2 \times 640 \times 480$ camera with up to 30 frames per second, inertial measurement unit (2 gyro meters and 3 accelerometers), 2 bumper sensors and 2 ultrasonic distance sensors.

In this research, we extend the perceptual system of the Nao to include a 3D motion capture capability implemented with the Kinect™ system. The Kinect recognizes a human body image in a configuration posture (see Fig. 3), and then continuously tracks the human body. Joint angles for three degrees of freedom in the shoulder and one in the elbow are extracted from the skeleton model, and mapped into the Nao joint space to allow real-time telecommand of the two arms.

D. iCub Humanoid and Reactable Perceptual System

The iCub is a 53 DOF humanoid platform developed within the EU consortium RobotCub. The iCub [44] is an open-source robotic platform with morphology approximating that of a 3(1/2) year-old child (about 104 cm tall), with 53 degrees of freedom distributed on the head, arms, hands and legs. The current work was performed on the iCubLyon01 at the INSERM laboratory in Lyon, France. The head has 6 degrees of freedom (roll, pan and tilt in the neck, tilt and independent pan in the eyes). Three degrees of freedom are allocated to the waist, and 6 to each leg (three, one and two respectively for the hip, knee and ankle). The arms have 7 degrees of freedom, three in the shoulder, one in the elbow and three in the wrist. The iCub has been specifically designed to study manipulation, for this reason the number of degrees of freedom of the hands has been maximized with respect to the constraint of the small size. The hands of the iCub have five fingers and 19 joints.

1) Motor Control: Motor control is provided by PMP. The passive motion paradigm (PMP) [45] is based on the idea of employing virtual force fields in order to perform reaching tasks while avoiding obstacles, taking inspiration from theories conceived by Khatib during 80s [46]. Within the PMP framework it is possible to describe objects of the perceived world either as obstacles or as targets, and to consequently generate proper repulsive or attractive force fields, respectively. A meaningful example of attractive force field that can be produced is the so called spring-mass-damper field; in this case the relevant parameters are the stiffness constant and the damping factor, which regulate the force exerted by a target placed in a given spatial location. An effective model that represents repulsive force fields is the multivariate Gaussian function, which accounts for a field centred at an obstacle and is characterized by the typical bell-shaped decay. According to the composition of all active fields, the manipulator's end-effector is eventually driven towards the selected target while bypassing the identified obstacles; evidently, its behavior and performances strictly depend on the mutual relationship among the tuneable field's parameters.

However, in order to tackle the inverse kinematics problem and compute the final trajectory of the end-effector, the original PMP makes use of the Transposed Jacobian algorithm; this method is well known to suffer from a number of weaknesses [47] such as the difficulty to treat constraints of complex kinematic structures as the iCub arm turns to be [48], [49]. Therefore, we have decided to replace the Transposed Jacobian approach with a tool that relies on a powerful and fast nonlinear optimizer, namely Ipopt [50]; the latter manages to solve the inverse problem while dealing with constraints that can be effectively expressed both in the robot's configuration space (e.g., joints limits) and in its task-space. This new tool [49] represents the backbone of the Cartesian Interface, the software component that allows controlling the iCub directly in the operational space, preventing the robot from getting stuck in kinematic singularities and providing trajectories that are much smoother than the profiles yielded by the first implementation of PMP.

In this changed context, the Cartesian Interface lies at the lowest level of the revised PMP architecture, whose simplified diagram is show in Fig. 3. At higher level the pmpServer element is responsible of composing the final force field according to the objects currently stored in an internal database. Users can add, remove or modify this database in the easiest way by forwarding requests to the server through a dedicated software interface, made available by the pmpClient component. It is important to point out that the properties of objects stored in the database can be retrieved for modification in real-time in order to mirror the environment as it evolves over time. All the software components of the revised PMP architecture can be openly accessed from the iCub repository.

2) Perception: In the current research we extend the perceptual capabilities of the iCub with the ReacTable™. The ReacTable is licensed by Reactable Systems. The ReacTable has a translucid surface, with an infrared illumination beneath the table, and detection system that perceives tagged objects on the table surface with an accuracy of ~ 5 mm. Thus, tagged objects can be placed on the table, and their location accurately captured by the infrared camera.

Interaction with the external world requires that the robot is capable of identifying its spatial reference frame with the objects that it interacts with. In the human being, aspects of this functionality is carried out by the dorsal stream, involving areas in the posterior parietal cortex which subserve complex aspects of spatial perception [51]. In our system, the 2D surface of the table is calibrated into the joint space of the iCub by a linear transformation calculated based on a sampling of three calibration points on the table surface that are pointed to by the iCub. Thus, three points are physically identified in the Cartesian space of the iCub, and on the surface of the ReacTable, thus providing the basis for calculation of a transformation matrix which allows the projection of object coordinates in the space of the table into the Cartesian space of the iCub. These coordinates can then be used as spatial arguments to the PMP action system of the iCub, described above, which provides basic physical actions including point-to(x, y, z), put(source X, Y, Z; target x, y, z), grasp(x, y, z), release(x, y, z).

IV. Multimodal Learning I: Nao & Kinect

Here we demonstrate the system functionality for real-time learning, in which language, spatial perception (via Kinect), and proprioception are combined to allow the Nao to learn a shared plan which is then executed in real-time with the human teacher. The idea is to solve a common problem encountered during daily life which is cleaning a table with some objects on it. As defined in the requirements analysis in Section II, the goal is to place the object from the table to a chest. However, the chest

needs to be opened before doing that, and we need to close it afterward to complete the task.

A. Learning the Shared Plan and Component Actions

Here we present the time-tagged unfolding of the cooperative dialog where the shared plan is established, and indicate how the system components are engaged to permit the corresponding interaction, and learning and execution of the shared plan.

1) Establishing the Shared Plan: The user begins by giving the name of the cooperative shared plan:

00:25 Nao: Tell me what I have to do.
00:26 Human: "Pedro, help me to clean the table."

The Nao recognizes the spoken order and checks in the shared plan manager if it knows this plan, and determines that there is no definition for such a plan. . .

00:30 Nao: "The shared plan is clean.
00:36 Nao: "Please explain me."

The shared plan manager creates an empty shared plan called *"clean the table"* and it is ready to be filled with the human description.

The user verbally explains all the steps of the shared plan, including both moves of the robot and himself. It is worth noting that the shared plan is initially specified in $\sim$10 s.

00:37 Human: "I reach the toy, then I reach the trash box then you open the trash box then I put the toy in the trash box then you close the trash box."

The Nao will segments the global shared plan in a sequential set of 3 actions using the grammar built in the spoken language interface, and then repeats it to be sure it has perfectly understood the whole interaction:

00:50: Nao: "I reach the toy, you reach the trash box, then I open the trash box, you put the toy in the trash box, I close the trash box. Do you agree with this plan?"

Two remarks should be made. First, note that in order to be intuitive and understandable, for relative pronouns "You" or "I" rather than repeating these literally, the Nao has created the plan by replacing the relative agent by the absolute ones: the human for "I" and the Nao for "You" because the human was speaking.

2) Correcting the Shared Plan: The second remark here is that there is in fact an error in the plan. Nao states that it should reach for the toy, rather than the human doing so. When the Nao is repeating the plan, it is not only to check if it has understood the orders but it also provides the human a way to verify his plan.

The user has here the possibility to edit the shared plan (Section II.A.2), and will do it by answering that he does not agree with this, for instance:

Human: "No I do not".

Thus the robot will ask where the problem is:

01:05 Nao: "So, What did I misunderstand?".

Here, the human has several ways to edit the plan:

— he could specify that there is a problem with a specific action by repeating it and could, after that, remove it or replace it by another one;
— he could erase everything and describe the plan again from scratch;
— he could insert a missing action.

In this dialogue example described below, the problem is with the agent of the first action. The Nao said "I reach the toy", but the intended action was "You reach the toy," with "you" referring to the human, so in response to the robot's question "what did I misunderstand?" the human names the erroneous action:

01:08 Human: "I reach the toy."

Thus, this dialog is motivated by the user's recognition that an error has occurred, followed by his specification of the erroneous component. This will trigger an editing of the plan, with the Nao asking

01:10 Nao: "What do you want me to do instead?"

A new command can now be inserted to replace the erroneous one.

01:13 Human: "I reach the toy."

This illustrates the detailed manipulation of relative pronouns, as the human is now speaking from his own perspective, and thus refers to himself as "I". To confirm the edit, the Nao then places a "virtual cursor" after the element to be replaced, and before the next action, and verifies that the new command should be inserted there:

01:15 Nao: "So where—I reached the toy, instead, you will reach the toy. Do you agree with this plan?
01:24 Human: "Yes, I do."

The robot now has a complete shared plan including tasks for the user and for itself. The next step is to verify that all of the tasks assigned to the robot are known.

3) Learning Component Actions: The Nao iterates through the shared plan until it encounters an action assigned to it, determines if it knows the action, and informs the user if not. A query is made to the Motor Primitive Library where the list of current known behaviors of the Nao is stored into. There may be several unknown behaviors in a new plan, thus to keep the interaction more animated the system selects one of the three interrogatory sentences at random.

Here, the Nao recognizes that it has to perform the "open" action, which is unknown for now and will call the human for help:

01:26 Nao: "Behavior 'open' is unknown. I am going to learn 'open'"

This will put the supervisor in the learning state. The human could create the behavior in the Motor Sequence Learning with one or a combination of these methods (see Fig. 2):

— calling a known behavior by language;
— asking the robot to imitate him;
— physically demonstrating the action by moving the robot's arms.

As illustrated in Fig. 4 (left column), the chest can be opened by pushing a lever, and the human will choose to teach "open"

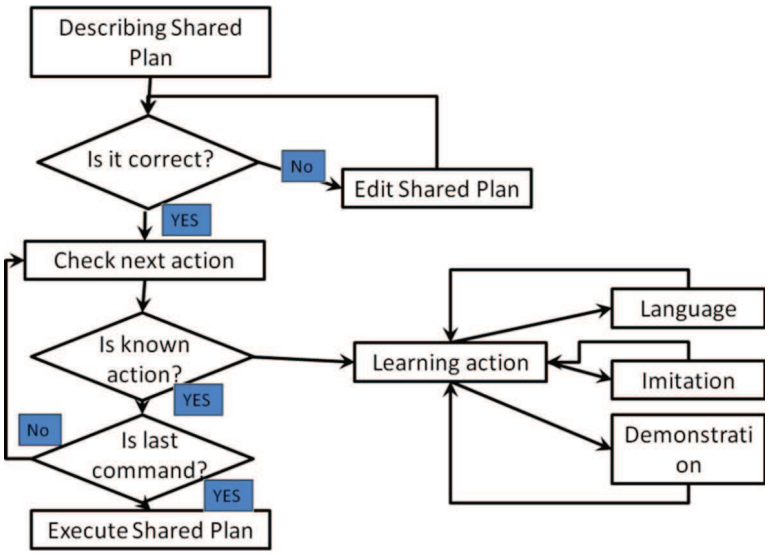


Fig. 2. Shared Plan Manager. In the initial state, the user describes the entire shared plan. The robot repeats the understood plan, allowing editing. Then, for each action, if it is unknown, the system enters specific states for learning by language, imitation or demonstration. When all actions are learned, the shared plan is executed.

by decomposing the action in two steps: First, the robot needs to move its arms up in the air, before, second, going down to push the lever. The "put-the-hands-up" is a known action, so the human will command this action.

```
01:34 Nao: "Tell me what to do."
01:36 Human:"Move to Kinect init."
```

This is recognized to be a known behavior in Motor Primitive Library, so the robot asks for a confirmation before performing the action:

```
01:38 Nao: "I am going to move to
Kinect init. Are you sure?"
01:42 Human: "Yes I am."
```

After the user's acknowledgment, Nao executes the behavior and the resulting joint angle trajectory will be stored in the new "open" behavior of motor sequence learning to begin the construction of this action.

The robot informs the user that the action is finished and it is waiting for the following command:

```
01:51 Nao: "Tell me what to do."
```

The human proceeds to the second step of the teaching by calling the imitation method:

```
01:53 Human:"Imitate me."
01:56 Nao: "Imitation activated. Tell
me when it is finished."
```

The system then enters the state (see Fig. 2) where joint angles are read from the Kinect device based on the observed human demonstrator (which is calibrated in the hands up position). The user then demonstrates the motion required to open the box, using visual feedback from the Nao in real-time. The corresponding joint angle trajectory is concatenated to those from the previous method. The user then terminates the imitation:

```
02:17 Human: "It is finished."
```

The "open" behavior has thus been learned by the Nao in real-time as a complex joint trajectory, and is transferred from Motor Sequence Learning to Motor Primitive Library so that it is now available as a command.

Following the flow outlined in Fig. 2, the Nao proceeds to the next behavior it should perform in the shared plan, and again there is an unknown action: "close":

```
02:20 Nao: "The behavior 'close' is
unknown. I am going to learn 'close.'"
02:27 Nao: "Tell me what to do."
```

An empty "close" behavior is created in motor sequence learning. This time, the human chooses to teach the action using the demonstration method:

```
02:29 Human: "Let me move your arms."
```

This places the supervisor in a state where it commands the Nao to set the stiffness of the arms to zero so that they can be manipulated by the human:

```
02:33 Nao: "Stiffness changed. Tell me
when it is finished."
```

As illustrated in Fig. 4 (right column) the human then physically demonstrates the action, pushing the Nao's hand and arm through a trajectory that closes the box. Once the demonstration is finished, the human terminates the learning, and asks the robot to proceed with the shared plan execution:

```
02:46 Human : "It is finished."
```

Thus, this dialog fragment, and Fig. 4, illustrate two complimentary learning methods: imitation of an action trajectory performed by the human, and physical demonstration where the human physically guides the learned trajectory of the robot. Depending on the nature of what is to be learned, the user can choose the more appropriate method.

B. Executing the Shared Plan

The shared plan is now fully specified, and the Nao begins execution, illustrated in Fig. 5.

```
02:56 Nao: "The current step is you
reach toy. You are supposed to reach.
Tell me when it is done."
03:01 Human: "It is done."
03:03 Nao: "The current step is I open
trash box."
```

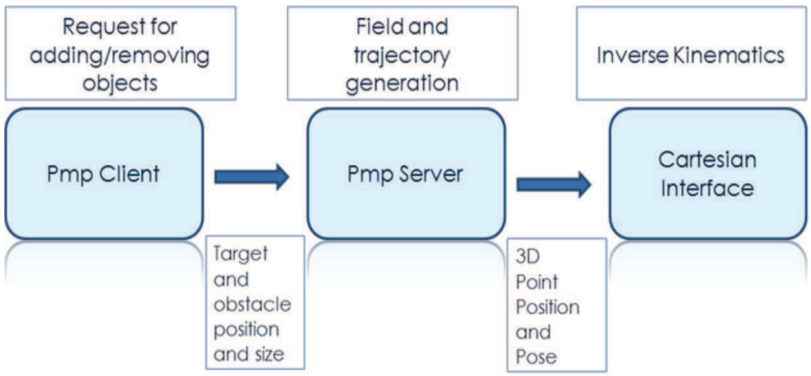


Fig. 3. PMP software architecture.

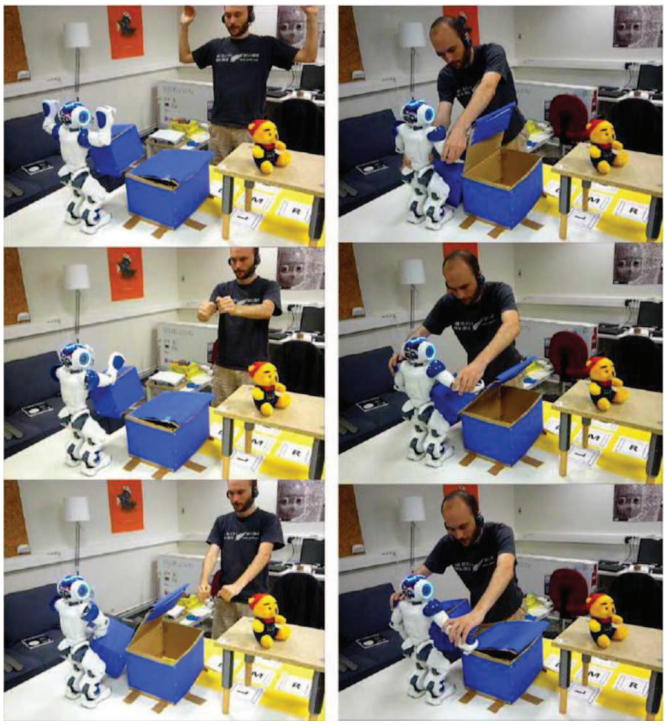


Fig. 4. Multimodal learning. (Left column) Using perceptual based imitation, the human demonstrates to the Nao how to open the box, and the Nao performs the action in real-time, mirroring the action of the human. (Right column) Using kinesthetic demonstration the human teaches the Nao how to close the box.

Here the robot uses the learned action trajectory to reproduce the "open" movement:

```
03:27 Nao: "The current step is you
put the toy trash box. You are sup-
posed to put. Tell me when it is
done."
03:34 Human: "It is done."
03:37 Nao: "The current step is I
close trash box."
```

The Nao then uses the joint trajectory that was taught by kinesthetic demonstration, to close the box. At this point the toy has successfully been closed in the box, and the shared plan is finished"

```
03:52 Nao: "This was the last com-
mand."
```

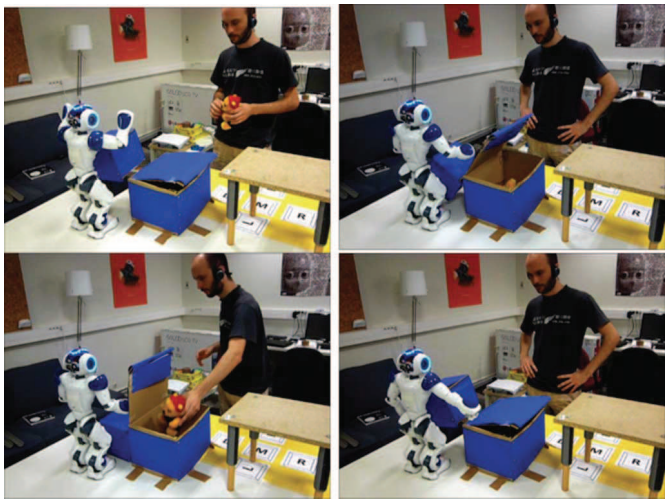


Fig. 5. Shared plan execution. Left column: Human takes toy, Nao opens box, human places toy in box. Right column: Nao closes box.

C. Performance Analysis

We analyze performance from three separate executions of the learning task described above. Two were performed in the laboratory, and the third was performed during the Robocup @home Open Challenge 2011 in Istanbul, July 2011. In this case, we were required to install and set up the system in 3 minutes, and then had five minutes to perform the task, with no possibility to shift to a different time, or to have another 5 minutes in case of failure. The task was successfully completed, and our "Radical Dudes" team placed 4th/19 in the Open Challenge. This demonstrates the robustness of the system.

For each of the three sessions where the shared plan was learned and then executed, we measured the time to complete the open-the-box and close-the-box actions during the learning phase, and then during execution of the learned shared plan. Execution time is measured from the onset of the human command, to the execution of the action and onset of next request by the Nao. Thus, during learning, the execution time includes the teaching component. In order to compare the effect of learning on the time to complete individual actions, we performed nonparametric Wilcoxon signed-rank test comparing each action when it was being learned vs. when it had been learned, collapsing across sessions. There were two actions per session (open and close), each performed once in learning and once in execution after learning. With the three sessions,

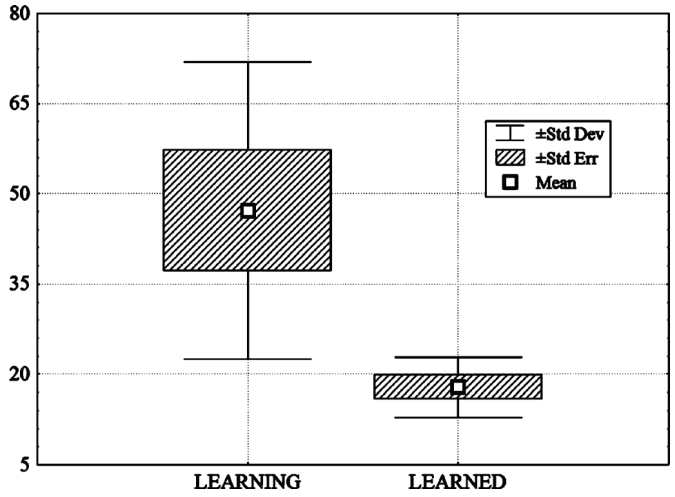


Fig. 6. Effects of shared plan learning on overall action execution time in seconds.

this provided a total of 6 learning-learned comparisons. As illustrated in Fig. 6, there is a significant reduction in execution time during the shared plan execution. This was confirmed in a significant learning effect in the Wilcoxon signed-rank test, $\mathrm{N} = 6, Z = 2.20, p = 0.027$. We thus demonstrated that the system can learn to produce arbitrary sequences of actions with a turn-taking structure. The principle limiting factor is simply the set of basic level actions from which the shared plans can be constructed. Three repetitions of the "clean-up" shared plan, including one during the Robocup@Home Open Challenge, demonstrate the reliability of the system. Over these three trials, we also demonstrated a significant effect of this learning (as opposed to simply commanding the robot) in terms of behavior execution time after learning.

D. Nao Experiment Discussion

We have previously demonstrated how the user can employ language to teach new actions [13], [14], and then combined the previously learned actions into a new shared plan [11], [15]. The current research extends this shared plan learning. For the first time, we demonstrate how spoken language can be used to coordinate online multimodal learning for a shared cooperative plan. The multiple modalities include imitation of actions performed by the human (using the Kinect), human demonstration of a desired trajectory by physically manipulating the robot arm, and finally, spoken language based invocation of known actions, with all of these modalities contributing to a coherent and integrated plan. We should stress that learning by imitation, demonstration and spoken language programming all have been extensively studied in the past. What is new here is the combination of these multiple modalities in a coherent and seamless manner, under the control of language. It is worth noting that while we emphasize the learning of the "clean the table" shared plan, the grammar-based learning capability allows for the construction of arbitrary turn-taking action sequences.

V. Multimodal Learning II: iCub & ReacTable

While the learning that we observed in the previous section has certain components that are platform specific (e.g., the morphology of the Nao, and the mapping of the Kinect to that morphology), the principal learning component which is based on spoken language is platform independent. Indeed, it provides a method for the linking together of action primitives into shared plans that can subsequently be used to achieve cooperative activity.

A. Learning the Shared Plan and Component Actions

In the first experiment with the iCub, the human explains a shared cooperative task similar to that used with the Nao, where the goal is to "hide" a toy under a box. The grammar is of the same structure as that used for the Nao. The principal difference is that it is constructed so that the plan is successively constructed from single actions that are concatenated with the previous actions (4):

1) $subjects = I | Me | You | Stephane | Maxime | Peter | iCub;
2) $objects = box | toy | trumpet | drums | left | middle | right;
3) $action =:
   a) $subjects [*sil%% | *any%%] grasp [*sil%% | *any%%]$objects|;
   b) $subjects [*sil%% | *any%%] point [*sil%% | *any%%]$objects|;
   c) $subjects [*sil%% | *any%%] put [*sil%% | *any%%] $objects
   d) $subjects [*sil%% | *any%%] uncover [*sil%% | *any%%] $ob
4) $sharedPlan = concatenate($sharedPlan, $action).

Here we illustrate how by exploiting this grammar, the shared plan manager can construct the shared plan from the sequence of actions described by the subject:

```
00:00 ICub: "What can I help you
with?"
00:02 Human: "Shared Plan."
00:05 ICub: "Shared Plan menu."
00:07 Human: "You and I hide the toy
with the box."
```

The Supervisor consults the Shared Plan Manager, and determines that there is no definition for this plan:

```
00:11 ICub: "Tell me how you and I can
hide toy box?"
00:13 ICub: "What is next?"
00:17 Human: "You put box left."
00:20 ICub: "Did you say 'you put box
left'?"
00:23 Human: "Yes I did."
00:25 ICub: "What is next?"
00:27 Human:"I put toy middle."
00:30 ICub: "Did you say 'I put toy
middle'?"
00:32 Human: "Yes I did."
00:35 ICub: "What is next?"
00:38 Human:"You put box toy."
00:40 ICub: "Did you say 'You put box
toy'?"
00:43 Human: "Yes I did."
00:46 ICub: "What is next?"
00:48 Human: "Finished."
```

The successive actions of the shared plan have now been extracted, and the plan is stored as

Hide(you, I, toy, box)
{(You put (box left))
(I put (toy middle))
(You put (box toy)).}

In the definition, the four arguments to hide are instantiated with the "you, I, toy, box." When the shared plan is used in the future, it can be called with different arguments, e.g., hide (John, you, trumpet, carton) for "John and you hide the trumpet with the carton," and these new arguments will be systematically substituted into the component actions of the shared plan. This provides the ability generalize to new objects and agents, including providing a capability for role reversal as specified in Section C.

B. Executing the Shared Plan

01:00 Human: "You and I hide the trumpet with the box."

Note here that the system allows arguments to be used in the "hide" shared plan. Here trumpet will be hidden.

01:04 ICub: "I will put the box on the left. You will put the trumpet on the middle. I will put the box on the trumpet. Should we proceed?"
01:24 Human: "Yes we should."
01:26 ICub: "I will put the box on the left.".

The robot performs its action, illustrated in Fig. 7(A)–(B).

02:02 ICub: "You will put the trumpet on the middle.".

The human makes the move, and the robot detects the change of position of the trumpet to the middle [see Fig. 7(C)].

02:12 ICub: "I will put the box on the trumpet.".

The iCub performs the final move, covering the trumpet with the box (Fig. 7(D)–(E)), completing the shared plan.

02:48 ICub: "Shared Plan menu.".

C. Role Reversal

In the cooperative task, the user can decide who takes which role. This implies the principal of role reversal. As stated in the introduction, role reversal is taken as evidence that the agent has a global representation of the actions that both agents will perform. This representation allows the agent to take on the role of either of the two participants in the shared plan [12].

In order to reverse roles, the human can use the same shared plan, and indicate role reversal by specifying the arguments in the reversed order. He can say "you and I hide the toy in the box" or "I and you hide the toy in the box." As specified in Section V-A above, depending on how the pronouns I and you are ordered in the specification, the roles of the human and robot will be reversed.

00:08 Human: "I and You will hide the toy in the box."

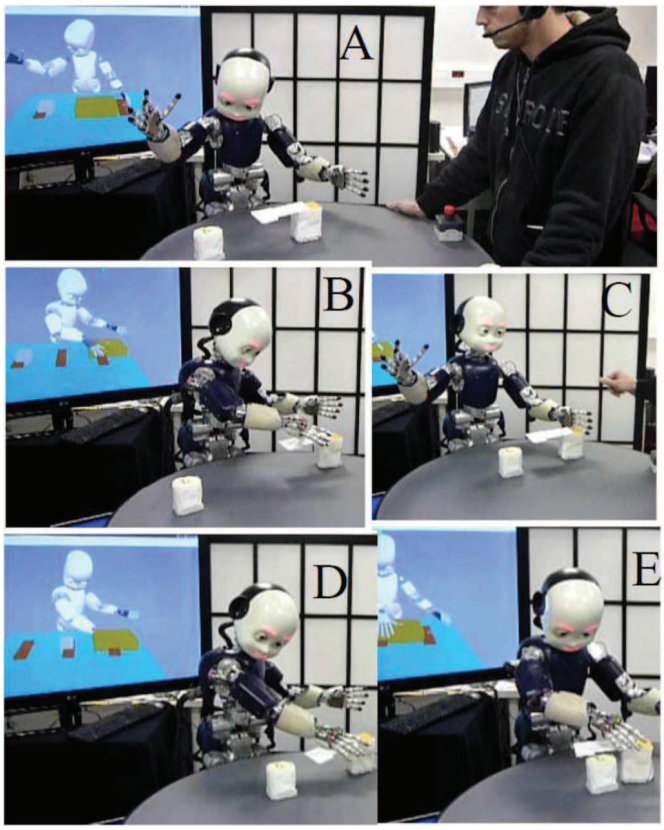


Fig. 7. Learning and performing the "hide the toy" cooperative plan. Setup with the "toy" on the left of the table, and "box" on the right. Spatial representation on iCub GUI left rear. B. iCub puts the box on its left. C. human put the toy in the middle. D. iCub reaches for the box, and F. puts the box on the toy. Note the grasping precision.

00:13 ICub: "You will put the box on the left. I will put the toy on the middle. You will put the box on the toy. Should we proceed?".
00:29 Human: "Yes we should."
00:31 ICub: "You will put the box on the left.".

Here the robot detects the change of position of the box to the left.

00:37 ICub: "I will put the toy on the middle.".
01:13 ICub: "You will put the box on the toy.".
01:19 ICub: "Shared Plan menu.".

Role reversal is a specific instance of a more general capability that is provided by the system. That is, once a shared plan has been learned with a given set of agent arguments, the arguments for the two agents can be instantiated with different instances, e.g., I and you vs. you and I.

D. Performance Analysis

We repeated the shared plan learning, execution and role reversal twice each. The timing of the principal events is illustrated in Fig. 8. It is noteworthy that the system allows the multiple-action shared plan to be specified in well-under one

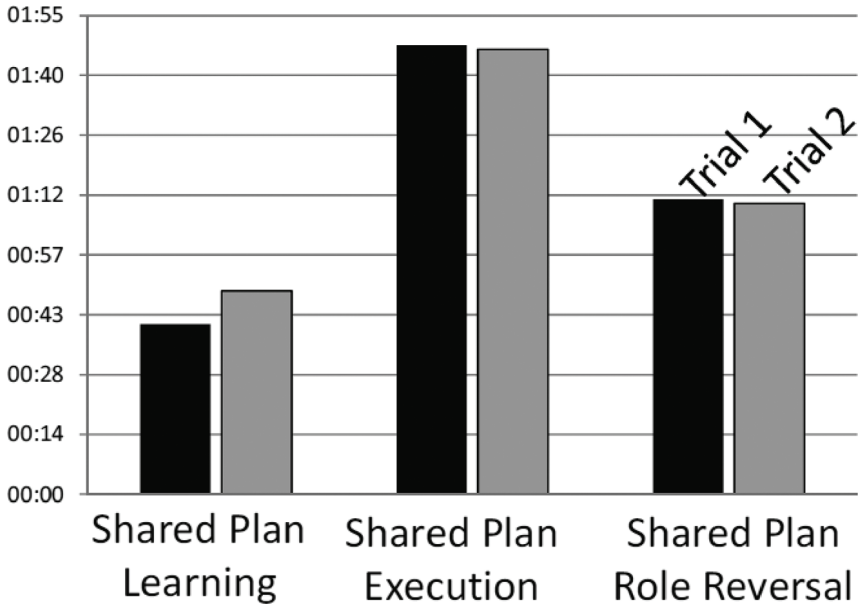


Fig. 8. Event durations (in seconds:minutes) for two trials (Trial 1 in black, trial 2 in grey) of the learning, execution and role reversal for the "hide the toy" shared plan.

minute, and then executed in this same time scale, both in the standard format, and the role reversal.

Note that in Fig. 8, the role reversal condition is executed more rapidly than the standard condition. This is due to the relative slowness of the robot actions, with respect to those of the human. In the standard sequence, the robot performs two actions (moving the box away from the center, and then over the toy) while the human performs only one action (placing the toy in the middle to be covered). This is reversed in the role, reversal, and thus the effect of the slowness of the robot is reduced.

E. iCub Discussion

These experiments extend the results with the Nao, which is in part achieved because of the more dexterous grasping capabilities of the iCub. In the current experiments we demonstrated how an arbitrary shared plan could be established in less than one minute, and then immediately be used to execute the cooperative task. In addition, we demonstrate how this shared plan can be used to allow role reversal, in which the two agents swap roles. Again, for Carpenter *et al.* [12] this is a hallmark of shared plan use, as it clearly demonstrates that the agents have a "bird's eye view" or global view, of the shared activity. Technically this requires that all of the actions that can take place in the shared plan can be executed physically by both the human and the robot. Because of the high spatial precision of the ReacTable, and the precision grasping capabilities of the iCub, this is a technical reality.

VI. Discussion and Future Work

The current research can be situated within the larger context of cognitive developmental robotics [52], with physical embodiment playing a central role in structuring representations within the system, through interaction with the environment, including humans. In development, the early grammatical constructions that are acquired and used by infants define structural mappings between the underlying structure of everyday actions, and the expression of this structure in language [6], [53]. We have exploited this mapping, in building systems that can learn grammatical constructions from experience with the environment [8], [54]. Here we exploit this type of grammatical construction, by building such constructions into the grammars that are used for speech recognition. These constructions that map onto the basic structure of action (e.g., *agent action object*) correspond to the basic argument constructions that are the workhorses of initial language [6], [53]. The "ditransitive" construction is a good example that has been extensively studied [5]. In a canonical form of this construction "`Subject Verb Recipient Object`" (e.g., John gave Sally a flower), Subject maps onto the agent of the transitive action specified by `Verb`, and `Recipient` receives the `Object` via that transitive action. The current research demonstrates how language, based on these constructions, can be used to coordinate real-time learning of cooperative actions, providing the coordination of multiple demonstration modalities including vision-like perception, kinesthetic demonstration [13], [29], [55]–[58], and command execution via spoken language. In this sense, language serves a dual purpose: First and most important, it provides the mechanism by which a cooperative plan can be constructed and modified. Second, during the construction of the shared plan, one of the modalities by which actions can be inserted into the plan is via the spoken issue of a command. We demonstrate that in this framework, the constructive features of language can be mapped onto different robot platforms. This requires the mapping of the argument structure of grammatical constructions onto the predicate-argument structure of the command and perceptual operators of the given platform [13], [55]. Doing so, we subsequently achieve performance, where the systems can learn and perform new cooperative behaviors in the time frame of

2–3 minutes. The introduction of structured language provides a powerful means to leverage sensory-motor skills into cooperative plans, reflecting how the development of language in human children is coincident with an explosion in their social development in the context of triadic relations between themselves, another person and a shared goal [1]. We should note that the "ecological validity" of the kind of language that the user can employ is somewhat restricted to simple grammatical constructions. That is, people cannot use fully unconstrained natural language, such as relative clauses, and pronouns. Still, this allows sufficient expressive ability for the user to construct elaborated shared plans.

The approach to learning that we have taken thus consists in the implementation of a highly structured scaffolding that allows the user to teach the robot new action components, and then to teach the robot how to organize these actions into more elaborate turn-taking sequences that constitute shared plans. The advantage of this approach is that it is powerful and scales well. It is powerful because it allows the user to specify arbitrary turn-taking sequences (which can even include solo sequences that are performed only by one of the agents), and the set of elementary actions can also be augmented through learning. All of this learning can be done with a single trial. The advantage of this is that learning is rapid. Indeed, related studies have demonstrated that for complex tasks such as those used here, human and neural network simulations fare better with high level instruction (imitation or verbal instruction) than with lower level instruction (reinforcement learning) [59]. The disadvantage is that the teaching must be perfect. Thus, in demonstrating a trajectory, the system cannot benefit from a successive refinement over multiple trials [60].

One of the limitations of this work is that there is not a systematic mechanism for the long-term accumulation and synthesis of such learning. In the future it will be important for these developmental acquisitions to be integrated into the system over a life-time scale [61]. Another limitation is that in the current research the behavior is determined by the shared plan, and there is no choice. To cope with changing task contingencies, the system will require more adaptive behavior including the ability to choose between competing options [62]. Perhaps one of the most fundamental limitations of the current research, which lays a foundation for future research, has to do with the deeper nature of the shared plan. This is the notion of the shared intention. Our robots can learn a plan that allows them to perform a cooperative task, and event to demonstrate role reversal. Yet the true notion of the actual final goal, the shared intention, to get that toy into the box, is currently not present. We have started to address this issue by linking actions to their resulting states, within the action representation [56]. We must go further, in order to now expand the language capability to address the expression and modification of internal representations of the intentional states of others.

The current research proposes an interaction architecture, for on-line multimodal learning, and demonstrates its functionality. It is not an extended user study that allows for the collection of data whose variability can be statistically analyzed in a population of subjects. Within the interactions that we test, the most pertinent parameter that reflects the change in the real-time flow and fluidity of the interactions is related to the time required for different component actions, and their changes as a function of learning. We thus demonstrate the feasibility of using spoken language to coordinate the creation of arbitrary novel turn-taking action sequences (which we refer to as shared plans). This includes the ability to create new actions (through demonstration and imitation), and to embed these actions in new turn-taking shared plans. Clearly a more robust demonstration of the performance of the architecture (and effective time gains before/after learning) should use naïve users and include metrics related to interaction quality, success etc. This is a topic of our ongoing research.

References

[1] M. Tomasello, M. Carpenter, J. Call, T. Behne, and H. Moll, "Understanding and sharing intentions: The origins of cultural cognition," *Behav. Brain Sci.*, vol. 28, pp. 675–691, 2005.

[2] S. Carey, *The Origin of Concepts*. Boston, MA, USA: MIT, 2009.

[3] F. Warneken, F. Chen, and M. Tomasello, "Cooperative activities in young children and chimpanzees," *Child Develop.*, vol. 77, pp. 640–663, 2006.

[4] F. Warneken and M. Tomasello, "Helping and cooperation at 14 months of age," *Infancy*, vol. 11, pp. 271–294, 2007.

[5] A. Goldberg, *Constructions: A Construction Grammar Approach to Argument Structure*. Chicago, IL, USA: Univ. Chicago Press, 1995.

[6] M. Tomasello, *Constructing a Language: A Usage Based Approach to Language Acquisition*. Boston, MA, USA: MIT Press, 2003.

[7] K. Rohlfing and J. Tani, "Grounding language in action," *IEEE Trans. Autonom. Mental Develop.*, vol. 3, p. 4, Dec. 2011.

[8] P. Dominey and J. Boucher, "Learning to talk about events from narrated video in a construction grammar framework," *Artif. Intell.*, vol. 167, pp. 31–61, 2005.

[9] P. Dominey, A. Mallet, and E. Yoshida, "Progress in programming the hrp-2 humanoid using spoken language," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2007, pp. 2169–2174.

[10] P. Dominey, A. Mallet, and E. Yoshida, "Real-time cooperative behavior acquisition by a humanoid apprentice," presented at the Int. Conf. Human. Robot., Pittsburg, PA, USA, 2007.

[11] P. Dominey and F. Warneken, "The basis of shared intentions in human and robot cognition," *New Ideas Psychol.*, vol. 29, p. 14, 2011.

[12] M. Carpenter, M. Tomasello, and T. Striano, "Role reversal imitation and language in typically developing infants and children with autism," *Infancy*, vol. 8, pp. 253–278, 2005.

[13] S. Lallée, S. Lemaignan, A. Lenz, C. Melhuish, L. Natale, S. Skachek, T. van Der Tanz, F. Warneken, and P. Dominey, "Towards a platform-independent cooperative human-robot interaction system: I. Perception," presented at the IROS, Taipei, Taiwan, 2010.

[14] S. Lallée, U. Pattacini, J. Boucher, S. Lemaignan, A. Lenz, C. Melhuish, L. Natale, S. Skachek, K. Hamann, J. Steinwender, E. A. Sisbot, G. Metta, R. Alami, M. Warnier, J. Guitton, F. Warneken, and P. F. Dominey, "Towards a platform-independent cooperative human-robot interaction system: II. Perception, execution and imitation of goal directed actions," in *Proc. IROS*, San Francisco, CA, USA, 2011, pp. 2895–2902.

[15] S. Lallée, U. Pattacini, S. Lemaignan, A. Lenz, C. Melhuish, L. Natale, S. Skachek, K. Hamann, J. Steinwender, E. A. Sisbot, G. Metta, J. Guitton, R. Alami, M. Warnier, T. Pipe, F. Warneken, and P. Dominey, "Towards a platform-independent cooperative human-robot interaction system: III. An architecture for learning and executing actions and shared plans," *IEEE Trans. Autonom. Mental Develop.*, vol. 4, no. 3, pp. 239–253, Sep. 2012.

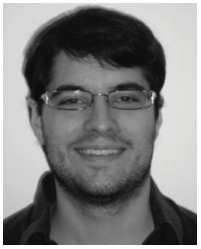
[16] H. H. Clark, "Coordinating with each other in a material world," *Discourse Studies*, vol. 7, pp. 507–525, Oct. 1, 2005, 2005.

[17] B. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robot. Autonom. Syst.*, vol. 57, pp. 469–483, 2009.

[18] M. Kaiser and R. Dillmann, "Building elementary robot skills from human demonstration," in *Proc. Int. Conf. Robot. Autom.*, 1996, pp. 2700–2705.

[19] M. Nicolescu and M. Mataric, "Natural methods for robot task learning: Instructive demonstrations, generalization and practice," in *Proc. 2nd Int. Joint Conf. Autonom. Agents Multiagent Syst.*, Melbourne, 2003, pp. 241–248.

[20] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robot. Autonom. Syst.*, vol. 57, pp. 469–483, 2009.
[21] P. Dominey, A. Mallet, and E. Yoshida, "Real-time spoken-language programming for cooperative interaction with a humanoid apprentice," *Int. J. Humanoids Robot.* , vol. 6, pp. 147–171, 2009.
[22] V. Tikhanoff, A. Cangelosi, and G. Metta, "Integration of speech and action in humanoid robots: iCub simulation experiments," *IEEE Trans. Autonom. Mental Develop.* , vol. 3, no. 1, pp. 17–29, Mar. 2011.
[23] Y. Zhang and J. Weng, "Task transfer by a developmental robot," *IEEE Trans. Evol. Comput.*, vol. 11, no. 2, pp. 226–248, Apr. 2007.
[24] P. Andry, P. Gaussier, S. Moga, J. P. Banquet, and J. Nadel, "Learning and communication via imitation: An autonomous robot perspective," *IEEE Trans. Syst., Man, Cybernetics, Part A: Syst. Humans*, vol. 31, no. 5, pp. 431–442, Sep. 2001.
[25] P. Andry, P. Gaussier, and J. Nadel, From Visuo-Motor Development to Low-Level Imitation 2002.
[26] C. A. Calderon and H. Hu, "Robot imitation from human body movements," presented at the AISB05 3rd Int. Symp. Imitation Animals Artifacts, 2005.
[27] S. Calinon, F. D'Halluin, E. L. Sauser, D. G. Caldwell, and A. G. Billard, "Learning and reproduction of gestures by imitation," *IEEE Robot. Autom. Mag.*, vol. 17, no. 2, pp. 44–54, Jun. 2010.
[28] K. Dautenhahn and C. L. Nehaniv, *The Correspondence Problem*. Cambridge, MA, USA: MIT Press, 2002.
[29] Y. Demiris and B. Khadhouri, "Hierarchical attentive multiple models for execution and recognition of actions," *Robot. Autonom. Syst.*, vol. 54, pp. 361–369, 2006.
[30] C. Breazeal and B. Scassellati, "Robots that imitate humans," *Trends Cogn. Sci.*, vol. 6, pp. 481–487, 2002.
[31] S. Schaal, "Is imitation learning the route to humanoid robots?," *Trends Cogn. Sci.*, vol. 3, pp. 233–242, 1999.
[32] M. Lungarella, G. Metta, R. Pfeifer, and G. Sandini, "Developmental robotics: A survey," *Connect. Sci.*, vol. 15, pp. 151–190, Dec. 01, 2003, 2003.
[33] J. Saunders, C. L. Nehaniv, and K. Dautenhahn, "Using self-imitation to direct learning," in *Proc. 15th IEEE Int. Symp. Robot Human Interact. Commun. (ROMAN)*, 2006, pp. 244–250.
[34] J. Saunders, C. L. Nehaniv, K. Dautenhahn, and A. Alissandrakis, "Self-imitation and environmental scaffolding for robot teaching," *Int. J. Adv. Robot. Syst.*, vol. 4, 2008.
[35] A. Cangelosi and T. Riga, "An embodied model for sensorimotor grounding and grounding transfer: Experiments with epigenetic robots," *Cogn. Sci.*, vol. 30, pp. 673–689, 2006.
[36] K. Coventry, A. Cangelosi, R. Rajapakse, A. Bacon, S. Newstead, D. Joyce, and L. Richards, , C. Freksa, M. Knauff, B. Krieg-Brückner, B. Nebel, and T. Barkowsky, Eds., *Spatial Prepositions and Vague Quantifiers: Implementing the Functional Geometric Framework Spatial Cognition IV. Reasoning, Action, Interaction*. Berlin, Germany: Springer-Verlag, 2005, vol. 3343, pp. 98–110.
[37] J. J. Steil, F. Röthling, R. Haschke, and H. Ritter, "Situated robot learning for multi-modal instruction and imitation of grasping," *Robot. Autonom. Syst.*, vol. 47, pp. 129–141, 2004.
[38] J. Weng, "Development a robotics: Theory and experiments," *Int. J. Humanoid Robot. (IJHR)*, vol. 1, pp. 199–236, 2004.
[39] P. F. Verschure, T. Voegtlin, and R. J. Douglas, "Environmentally mediated synergy between perception and behaviour in mobile robots," *Nature*, vol. 425, pp. 620–624, Oct. 9, 2003.
[40] A. Duff, M. S. Fibla, and P. F. Verschure, "A biologically based model for the integration of sensory-motor contingencies in rules and plans: A prefrontal cortex based extension of the distributed adaptive control architecture," *Brain Res. Bull.*, vol. 85, pp. 289–304, June 30, 2011.
[41] P. F. Verschure and T. Voegtlin, "A bottom up approach towards the acquisition and expression of sequential representations applied to a behaving real-world device: Distributed adaptive control III," *Neural Netw.*, vol. 11, pp. 1531–1549, Oct. 1998.
[42] S. Sutton, R. Cole, J. Villiers, J. Schalkwyk, P. Vermeulen, M. Macon, Y. Yan, E. Kaiser, B. Rundle, and K. Shobaki, "Universal speech tools: The CSLU toolkit," presented at the 5th Int. Conf. Spoken Language Process., 1998.
[43] P. Fitzpatrick, G. Metta, and L. Natale, "Towards long-lived robot genes," *Robot. Autonom. Syst.*, vol. 56, pp. 29–45, 2007.
[44] G. Metta, G. Sandini, D. Vernon, L. Natale, and F. Nori, "The iCub humanoid robot: An open platform for research in embodied cognition," in *PerMIS: Performance Metrics for Intelligent Systems Workshop*, Washington, DC, USA, 2008, pp. 19–21.
[45] V. Mohan, P. Morasso, G. Metta, and G. Sandini, "A biomimetic, force-field based computational model for motion planning and bimanual coordination in humanoid robots," *Autonom. Robots*, vol. 27, pp. 291–307, 2009.
[46] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *Int. J. Rob. Res.*, pp. 90–98, 1986.
[47] L. Sciavicco and B. Siciliano, Modelling and Control of Robot Manipulators 2005.
[48] A. Parmiggiani, M. Randazzo, L. Natale, G. Metta, and G. Sandini, "Joint torque sensing for the upper-body of the iCub humanoid robots," presented at the IEEE Int. Conf. Humanoid Robots, Paris, France, 2009.
[49] U. Pattacini, F. Nori, L. Natale, G. Metta, and G. Sandini, "An experimental evaluation of a novel minimum-jerk cartesian controller for humanoid robots," presented at the IROS, Taipei, Taiwan, 2010.
[50] A. Wächter and L. T. Biegler, "On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming," *Math. Program.*, vol. 106, pp. 25–57, 2006.
[51] L. Shmuelof and E. Zohary, "Dissociation between ventral and dorsal fMRI activation during object and action recognition," *Neuron*, vol. 47, pp. 457–470, 2005.
[52] M. Asada, K. Hosoda, Y. Kuniyoshi, H. Ishiguro, T. Inui, Y. Yoshikawa, M. Ogino, and C. Yoshida, "Cognitive developmental robotics: A survey," *IEEE Trans. Autonom. Mental Develop.*, vol. 1, no. 1, pp. 12–34, May 2009.
[53] A. Goldberg, *Constructions: A Construction Grammar Approach to Argument Structure*. Chicago, IL, USA: Univ. Chicago Press, 1995.
[54] P. Dominey and J. Boucher, "Developmental stages of perception and language acquisition in a perceptually grounded robot," *Cogn. Syst. Res.*, vol. 6, pp. 243–259, 2005.
[55] S. Lallée, S. Lemaignan, A. Lenz, C. Melhuish, L. Natale, S. Skachek, T. van Der Tanz, F. Warneken, and P. Dominey, "Towards a platform-independent cooperative human-robot interaction system: II. Perception, execution and imitation of goal directed actions," presented at the IROS, 2011, presented at the.
[56] S. Lallée, C. Madden, M. Hoen, and P. Dominey, "Linking language with embodied teleological representations of action for humanoid cognition," *Frontiers Neurobot.*, 2010.
[57] S. Lallée, F. Warneken, and P. Dominey, "Learning to collaborate by observation," presented at the Epirob, Venice, Italy, 2009.
[58] G. Metta, G. Sandini, L. Natale, L. Craighero, and L. Fadiga, "Understanding mirror neurons: A bio-robotic approach," *Interact. Stud.*, vol. 7, pp. 197–232, 2006.
[59] F. Dandurand and T. R. Shultz, "Connectionist models of reinforcement, imitation, and instruction in learning to solve complex problems," *IEEE Trans. Autonom. Mental Develop.*, vol. 1, no. 2, pp. 110–121, Aug. 2009.
[60] S. Calinon, F. Guenter, and A. Billard, "On learning, representing, and generalizing a task in a humanoid robot," *IEEE Trans. Syst. Man. Cybern. B Cybern.*, vol. 37, pp. 286–298, Apr. 2007.
[61] Y. Demiris and A. Meltzoff, "The robot in the crib: A developmental analysis of imitation skills in infants and robots," *Infant Child Develop.*, vol. 17, pp. 43–53, Jan. 2008.
[62] T. J. Prescott, F. M. Montes Gonzalez, K. Gurney, M. D. Humphries, and P. Redgrave, "A robot model of the basal ganglia: Behavior and intrinsic processing," *Neural. Netw.*, vol. 19, pp. 31–61, Jan. 2006.



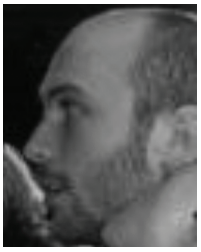
Maxime Petit received the M.Sc. degree in computer sciences (cognitive sciences specialty) from the University of Paris-Sud, Orsay, France, in 2010, and an engineering degree in biosciences (bioinformatics and modeling specialty) from the National Institute of Applied Sciences (INSA), Lyon, France, the same year. He is currently working towards the Ph.D. degree at the Stem-Cell and Brain Research Institute, from the unit 846 of the National Institute of Science And Medical Research (INSERM) in Bron, France, working in the Robotic Cognition Laboratory team. He is focusing about the reasoning and planning in robotics, especially with the iCub platform, and more precisely, within a context of spoken language interaction with a human.

Stéphane Lallée received the Master's degree in cognitive science and human–machine interface engineering from the University of Grenoble, Grenoble, France, in 2008. He received the Ph.D. degree in cognitive neuroscience from Lyon University, Lyon, France, in 2012, developing a distributed architecture for human–robot cooperation.

He joined the Robot Cognition Laboratory in 2008, and has played a leading role in the iCub project in Lyon since the arrival of the iCub in 2009.

Jean-David Boucher received the Ph.D. degree in cognitive science and robotics from the the Robot Cognition Laboratory, University of Lyon, Lyon, France, in 2010.

He continued there as a Postdoctoral Fellow on the CHRIS project. He currently teaches programming at Monash University, Melbourne, Australia, and pursues research on human–robot interaction.



Grégoire Pointeau received the M.Sc. degree in biosciences (bioinformatics and modeling specialty) from the National Institute of Applied Sciences (INSA), Lyon, France, in 2011, and the Master's degree in cognitive science at the University of Lyon, Lyon, France. He is currently working toward the Ph.D. degree at the Robot Cognition Laboratory in Lyon as part of the FP7 EFAA project.

He is interested in the accumulation of experience and autobiographical memory in the elaboration of the self, in the context of human–robot social interaction.

Pierrick Cheminade received the Technical Dipl. in computer sciences from the University Institute of Technology, Puy-en-Velay, France, in 2008.

Before working in the game industry he was an independent web developer. He began his career in 2008 at DreamonStudio Lyon, France, and then worked for two years at Little World Studio (2008–2011 Lyon France) before working at the Robot Cognition Laboratory, on the Nao, at the Inserm-U846 (2011). In 2012 he released several games as an independent on the Ios platform. He is currently working as a game programmer for Codemaster (Southam United Kingdom).

Dimitri Ognibene received the B.Sc. degree computer engineering from the University of Palermo, Palermo, Italy, in 2004, and the Ph.D. degree in robotics from the University of Genoa, Genoa, Italy, in 2009.

He was Research Assistant the CNR ISTC in Rome from 2005 until 2011. In 2010, he was visiting scholar at the University of Massachussets Amherst, Amherst, MA, USA. He is currently a Research Associate at Imperial College London, London, U.K. His research interests include the computational principles of visual attention, resource allocation, and learning and development in natural and artificial systems. He is now working on probabilistic social attention systems for humanoid robots.

Eris Chinellato (M'03) received the B.Sc. degree in industrial engineering from the Università degli Studi di Padova, Padova, Italy, in 1999, the M.Sc. degree in artificial intelligence from the University of Edinburgh, Edinburgh, U.K., in 2002, and the Ph.D. degree in intelligent robotics from Jaume I University, Spain, in 2008.

He is now working at Imperial College London on neural models of social interaction to be applied to the iCub humanoid robot. He has published in influential journals and proceedings in both robotics and neuroscience, and has served as reviewer and program committee member for several IEEE journals and conferences. His interdisciplinary research, integrating robotics with experimental and theoretical neuroscience, focuses on sensorimotor integration in both natural and artificial systems.

Ugo Pattacini received the M.S. degree (with Hons.) in electronic engineering from University of Pisa, Pisa, Italy, in 2001, and the Ph.D. degree in robotics, neurosciences, and nanotechnologies from Istituto Italiano di Tecnologia (ITT), Genova, Italy, in 2011.

He is currently a Postdoctoral Fellow in the Robotics, Brain, and Cognitive Sciences Department (RBCS) at IIT. From 2001 to 2006, he worked as an Embedded Software Developer for Formula 1 applications first at Racing Department of Magneti Marelli in Milan and then joining Toyota F1 team in Cologne, dealing with the design and the implementation of proprietary hard-constrained real-time operating systems, vehicle dynamics and torque-based traction control strategies. From 2006 to 2007, he moved to Earth Observation Business Unit of Thales Alenia Space in Rome where he was concerned with the specifications design and trade-off analyses of the satellite data acquisition system for COSMO-SkyMed (ASI) and GMES (ESA) scientific program. From 2008, he is involved in the development of the humanoid iCub at IIT focusing his interests mainly on the advancement of robot motor capabilities and pursuing a methodology that aims to combine traditional model-based approaches with the most recent machine learning techniques.

Ilaria Gori was born in Rome, Italy, on November, 6th 1986. She received the Bachelor's degree in management engineering with the score of 110/110 at Sapienza, University of Rome, Rome, Italy, in July 2008. In 2010, she graduated (*cum laude*) in computer engineering at Sapienza, University of Rome, with a specialization in artificial intelligence.

She wrote her Master's thesis at Imperial College, in London, where she lived for six months in 2010. Then she worked for two months in a software development company in 2010. In January 2011, she started a Ph.D. in robotics, cognition and interation technologies at Istituto Italiano di Tecnologia, Genova, Italy, that she is currently carrying out. She is mainly interested in computer vision and machine learning, but she also worked in robotics.

Ms. Gori published a paper at ISVC 20 summarizing her Master's thesis, and she won a Best Paper Award.

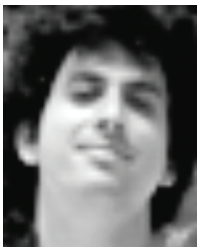
Uriel Martinez-Hernandez is currently working towards the Ph.D. degree in automatic control and systems engineering at the University of Sheffield, Sheffield, U.K.

His current research is about exploration and object recognition based on active touch sensing using the fingers of the ICUB robot. Also in this project, he is working together with the Psychology department.

Hector Barron is an Engineer with a first degree in computer science. He worked several years in applying computer vision and machine learning upon real-time systems. Attracted by cognitive science, he is proposing increasing the capacities of spatial reasoning for navigation tasks. Upon the Bayesian framework, this work explores interesting insights from neuroscience and psychology.

Martin Inderbitzin received the Ph.D. degree in computer science and robotics in the Synthetic, Perceptive, Emotive, and Cognitive Systems (SPECS) Group, in Barcelona.

He studied Embodied Models of Emotions—Verification of Psychological and Neurobiological Theories of Emotions Using Virtual and Situated Agents. He is currently R&D Director Projektil Pot Shot Experience Design, Zürich, Switzerland, where he will investigate video mapping and interactivity developing embodied experiences.



Andre Luvizotto received the B.Sc. degree in music, 2005, and athe M.Sc. degree in electrical engineering, in 2008, both in UNICAMP, with the cooperation of NICS—" Núcleo Interdisciplinar de Comunicação Sonora." In his Master's thesis, he worked with sonological models, based on wavelet transform. He is currently working towards the Ph.D. degree at the SPECS group at the Universitat Pompeau Fabra, Barcelona, Spain.

He joined SPECS in February, 2008. He is currently working on the EFAA project, with a computational model for visual and auditory invariant representations, based on temporal population code.

Vicky Vouloutsi received the B.A. degree in computer science from the Technological Educational Institute of Athens, Athens. Greece, 2008. In 2009, she came to Barcelona where she received the M.Sc. degree in cognitive systems and interactive media from the Universitat Pompeu Fabra, in 2011. She completed her Master's thesis on biologically inspired computation for chemical sensing in SPECS where she is currently working as a continuation of her Master's thesis.

Yiannis Demiris (M'01–SM'08) received the B.Sc. and Ph.D. degrees in intelligent robotics from the University of Edinburgh, Edinburgh, U.K., in 1994 and 1999, respectively.

He joined the faculty of the Department of Electrical and Electronic Engineering at Imperial College London in 2001 where he is currently a Reader in human-centred robotics and heads the Personal Robotics Laboratory. His research interests include biologically inspired robotics, human-robot interaction, developmental robotics, and robotic assistive devices for adults and children with disabilities. He was the Chair of the IEEE International Conference on Development and Learning in 2007 and the Program Chair of the ACM/IEEE International Conference on Human-Robot Interaction in 2008. He has organized several international workshops on robot learning, bioinspired machine learning, and epigenetic robotics.

Dr. Demiris has received fellowships from the AIST-MITI in Japan, and the European Science Foundation, and currently participates in several EU FP7 research projects in Human–Robot Interaction. In 2012, he received the Rector's Award and the Faculty of Engineering Award for Excellence in Engineering Education.

Giorgio Metta received the M.Sc. (with Hons) and Ph.D. degrees in electronic engineering from the University of Genoa, Genoa, Italy, in 1994 and 2000, respectively.

He is currently a Senior Scientist at IIT and Assistant Professor at the University of Genoa where he teaches courses on anthropomorphic robotics and intelligent systems for the bioengineering curricula. From 2001 to 2002, he was Postdoctoral Associate at the MIT AI-Lab where he worked on various humanoid robotic platforms. He has been an Assistant Professor at the University of Genoa since 2005 and with IIT since 2006. His research activities are in the fields of biologically motivated and humanoid robotics and in particular in developing life-long developing artificial systems that show some of the abilities of natural systems. His research developed in collaboration with leading European and international scientists from different disciplines including neuroscience, psychology, and robotics. Giorgio Metta is author or coauthor of approximately 100 publications. He has been working as research scientist and co-PI in several international and national funded projects. He has been reviewer for international journals, and national and international funding agencies.

Peter Ford Dominey received the B.A. degree in cognitive psychology and artificial intelligence from Cornell University, Ithaca, NY, USA, in 1984 . He received the M.Sc. and Ph.D. degrees in computer science from the University of Southern California, Los Angeles, CA, USA, in 1989 and 1993, respectively, developing neural network models of sensorimotor sequence learning, including the first simulations of the role of dopamine in sensorimotor associative learning, and pioneering work in reservoir computing.

He is currently a CNRS Research Director at the INSERM Stem Cell and Brain Research Institute in Lyon, France, where he directs the Robot Cognition Laboratory. From 1984 to 1986, he was a Software Engineer at the Data General Corporation, and from 1986 to 1993 he was a Systems Engineer at NASA/JPL/CalTech. In 1997 he became a tenured researcher, and in 2005 a Research Director with the CNRS in Lyon France. His research interests include the development of a "cognitive systems engineering" approach to understanding and simulating the neurophysiology of cognitive sequence processing, action and language, and their application to robot cognition and language processing, and human–robot cooperation. He is currently participating in several French and European projects in this context.

Chapter 5

Reasoning Based on Integrated Real World Experience Acquired by a Humanoid Robot

5.1 Introduction

This publication presents an implementation of the autobiographical memory, using both episodic and semantic memory, which store respectively personal events with precise objects, localization, actions, ... and general knowledge or facts about time, spatial or contextual items ([Tulving et al., 1988, Conway and Pleydell-Pearce, 2000, Cohen and Conway, 2007]). Through interaction with Human, the robot will acquire data and build his own knowledge based on his experience, filling in an SQL database. Inspired by the teleological stance taken by children ([Gergely and Csibra, 2003]), we will investigate how we can extract pre-conditions or effects of observed actions in order to be able to understand and manipulate them, allowing the iCub the be able to reason, planify and solve unseen problems.

5.2 Publication

Reasoning Based on Integrated Real World Experience

Acquired by a Humanoid Robot

Maxime Petit, Gregoire Pointeau, Peter Ford Dominey*

Robot Cognition Laboratory, INSERM U846, Bron France

maxime.petit@inserm.fr, gregoire.pointeau@inserm.fr, peter.dominey@inserm.fr

Inserm U846
Stem Cell and Brain Research Institute
18, avenue du doyen Jean Lépine
69675 Bron cedex
France
Tél : +33-4 72 91 34 84
Fax : +33-4 72 91 34 61

Keywords: humanoid robot; embodied; physical interaction; experience; reasoning; planning; PDDL; autobiographical memory; learning; human-robot interaction.

Manuscript statistics:

Words: 10,000

Figures: 12

Tables: 4

Pages: 40

Abstract.

One of the long-term strengths of research in artificial intelligence has been the development of reasoning systems that can exploit expert knowledge in well-defined task domains. A non-trivial problem in this domain is getting information coded in the knowledge representation. For example, as in human development, the acquisition of knowledge at one level requires the consolidation of knowledge from a lower level. How is accumulated experience structured so as to allow the individual to apply this structured knowledge to new situations? The current research investigates how a robotic system that interacts with humans can acquire knowledge that can be formalized automatically, forming the expert knowledge that can be used for reasoning. Through physical interaction with a human, the iCub robot acquires experience about spatial locations. Once consolidated, this knowledge can be used in further acquisition of experience concerning the preconditions and consequences of actions. Finally, this knowledge can be translated into rules that can be used for reasoning and planning in novel problem solving situations. We demonstrate how multiple levels of knowledge acquisition are organized, based on experience in interaction with humans, in two distinct problem solving domains. In the more complex domain, we demonstrate how the robot can learn the rules of the Tower of Hanoi and solve novel instances of the problem, without ever having seen a complete solution. This research illustrates how real world knowledge can be acquired by robots for use in AI planning and reasoning systems. This can provide the first step for more flexible systems that can avoid the brittleness that has sometimes been associated with traditional AI solutions where knowledge has been pre-specified.

1 Introduction

The ability to reason can be considered to rely on two complimentary components. First, a system must have access to some form of knowledge or expertise from which it can reason. Second, the system must then have some form of reasoning capability that allows that knowledge to be used in a systematic way [1]. Within this context, the current research attempts to determine how experience

that a robot can acquire through interaction with a human can be used as the basis for knowledge-based reasoning. Allowing robots to learn from experience has been a long-term goal in cognitive robotics [2]. Spoken language has been used to "program" robots, that is, to specify procedures for how to achieve tasks, including navigation [3], interaction [4-8] and more elaborate shared plans for joint cooperative action with the human [9-13]. In this framework, knowledge is transferred from the human to the robot in a direct manner. Ideally, the robot should be able to extract information from its experience rather than having the knowledge explicitly specified. This knowledge could then be used order to allow the robot to reason about new situations. Via this approach, robotics research could make contact with symbolic reasoning in AI.

The symbolic reasoning capability of AI is built around three basic capabilities: how to represent knowledge so that the system can use this knowledge for problem solving; the actual problem solving achieved by inference machines; and control of exponential complexity in the domain [1]. In addition there are requirements for supplemental knowledge about the domain, and all of this must be encompassed in a coherent architecture. A proven AI approach can use rules for representation, forward and backward chaining for inference, goal directed reasoning for control, and a rule based problem solving architecture [1].

We focus on the knowledge generation and representation component, and in particular, how knowledge about the domain can be acquired in an autonomous manner. In traditional AI [1], knowledge engineers would elicit expert knowledge from experienced people, and codify it so that this expert knowledge would allow the system to reason. In the modern context of adaptive cognitive systems for robots, the goal is to allow the robot itself to become an expert by accumulating knowledge from its own experience [14-16].

From the outset of development, the infant begins to extract regularities from the environment, and in a recursive manner to further extract structure based on this growing repertoire [17, 18]. The goal of this research is to provide a real-time goal-directed reasoning capability to robots, loosely based on a development-like trajectory where knowledge from successively refined levels contributes to the ability to reason. When we reach the highest level of this representational hierarchy, the information will be appropriate for reasoning in the AI sense. Based on the Planning Domain Definition Language (PDDL) format [19, 20], actions encountered by the robot and stored in its AutoBiographical-like Memory (ABM) are statistically processed in order to extract contextual knowledge

about them [21]. This knowledge is formatted in PDDL allowing an AI planner to produce the sequence of actions allowing the robot to fulfill a goal. We will demonstrate the system capabilities with two concrete example tasks, where the robot learns the physical structure of the environment, the rules of the tasks, and then demonstrates its ability to use this knowledge to reason in novel situations. The first task or game involves rules about how objects can be displaced from one location to another in terms of pre- and post-conditions for actions. The second game is more involved, based on the Tower of Hanoi, adapted to our environment and here called the "Table of Hanoi". In this context the system must learn about the particular properties of objects in terms of when and where they can move, based on the presence of other objects at the source and target destinations.

********** Figure 1 about here **********

2 Robot System Description

We first describe the global architecture of the robot and control system, and then provide a more detailed description of the memory and reasoning systems. The human-robot interaction set-up is illustrated in Figure 1. The iCub robot interacts with humans, using an interactive table that allows for precise object localization. As seen in the figure, the Graphical User Interface (GUI) displays the robot and the positions of recognized objects on the table. An overview of the system architecture is also illustrated in Figure 1, and explained, element by element, below. Part of the core basis of our research is the implementation of a multi-level memory system, illustrated in Figure 2.

********** Figure 2 about here **********

Following Figure 2, direct perceptual experience is represented in the episode-like memory (ELM), and through the detection and extraction of recurring regularities (e.g. the word "left" and spatial coordinates within a certain dispersion ellipse), a higher level semantic memory is generated through a process we refer to as consolidation (inspired by the same term in animal physiology). This accumulated knowledge (e.g. the meaning of the term "left") can lead to a reinterpretation of past experiences in the ELM, in a process that we refer to as "retro-reasoning". Using this knowledge

about spatial locations, the system can begin to extract regularities about pre-and post-conditions that hold before and after acting, respectively. This information can be transformed into a format compatible for AI planners, thus allowing the robot to use its accumulated experience to reason about new situations. Given this overview, we can now describe the system in more detail.

2.1 iCub

The current work was performed on the iCubLyon01 at the INSERM Robot Cognition Laboratory in Lyon, France. The iCub is a 53 DOF humanoid platform developed within the EU consortium RobotCub. The iCub [22] is an open-source robotic platform with morphology approximating that of a 3½ year-old child (about 104cm tall), with 53 degrees of freedom distributed on the head, arms, hands and legs. The head has 6 degrees of freedom (roll, pan and tilt in the neck, tilt and independent pan in the eyes). Three degrees of freedom are allocated to the waist, and 6 to each leg (three, one and two respectively for the hip, knee and ankle). The arms have 7 degrees of freedom, three in the shoulder, one in the elbow and three in the wrist. The iCub has been specifically designed to study manipulation, for this reason the number of degrees of freedom of the hands has been maximized with respect to the constraint of the small size. The hands of the iCub have five fingers and 19 joints. Motor control for the robot requires identification of object locations in space, and computation of the required joint trajectories [23]. As part of the iCub software architecture the YARP communication protocol is used throughout the system, in order to allow well defined port-based client-server connections between the different components described below.

2.2 ReacTable

In order to allow high precision perception of objects, both for understanding scenes and events, as well as for allowing precise and reliable reaching and grasping, we have adopted the ReacTable™ interactive table. The ReacTable has a translucid surface, with an infrared (IR) illumination and IR camera detection system beneath the table that perceives tagged objects on the table surface with an accuracy of ~5mm. Thus, tagged objects can be placed on the table, and their location accurately captured by the IR camera inside the table.

Interaction with the external world requires that the robot is capable of identifying its spatial reference frame with the objects that it interacts with. This is similar to the human, where aspects of this functionality is carried out by the dorsal visual stream, involving areas in the posterior parietal cortex which subserve complex aspects of spatial perception [24]. In our system, the 2D surface of the table is calibrated into the joint space of the iCub by a linear transformation calculated based on a sampling of four calibration points on the table surface that are pointed to by the iCub. Thus, four points are physically identified in the Cartesian space of the iCub, and on the surface of the ReacTable, thus providing the basis for calculation of a transformation matrix which allows the projection of object coordinates in the space of the table into the Cartesian space of the iCub. These coordinates can then be used as spatial arguments to the action system of the iCub, described below, which provides basic physical actions including point-to(x, y, z), put(source x, y, z; target x, y, z), grasp(x, y, z), release(x, y, z). In the current experiments, all objects can be grasped with the same grasp parameters, so these are not independently specified.

2.3 Object Properties Collector

The common space in which the human and robot interact with objects is on the surface of the ReacTable. The current state of the world, in terms of those objects, the human and the iCub, is stored in the Object Properties Collector (OPC) which thus contains all the information about objects, agents, entities or relations. The OPC can be considered as the mental representation of the Robot. For example, all the information gathered by the ReacTable, or any other sensor will be stored in real time in the OPC.

The ReacTable2OPC client receives the data about the real-time position of objects from the ReacTable software and stores this in the OPC. The data acquired from each object once on the table includes: a unique ID, position with respect to the coordinate frame of the table, angle, speed, rotation and whether the object are still present on the table or not, as stated above. As stated above, in order to allow coherent interaction, the iCub and the ReacTable are calibrated into a common physical space that is based on the egocentric frame of the iCub.

2.4 Interaction Supervisor

The Supervisor (Figures 1 and 3) provides the general management function for the human-robot interaction, and is implemented using a state-based dialog management capability. This allows the user to enter different interaction states related to teaching spatial location, action and temporal primitives and shared plans. The Supervisor function is implemented with the CSLU Rapid Application Development (RAD) Toolkit [25], a state-based dialog system which combines state-of-the-art speech synthesis (Festival) and recognition (Sphinx-II) in a GUI programming environment. RAD allows scripting in the TCL language and permits easy and direct binding to the YARP domain, so that all access from the Supervisor function with other modules in the architecture is via YARP.

We have previously explored how the argument structure of sentences (e.g. "Put the circle on the left") allows for a structured mapping onto the argument structure of perceptual and motor commands for robots, and we use such mappings here [5, 6, 13, 26, 27] in the Supervisor.

********** Figure 3 about here **********

3 Autobiographical Memory and Reasoning

A central aspect of this research is that the experience of the robot shall be captured in a structured, time ordered record, and that this record can be used to generate appropriate behavior in the future. We implement an autobiographical memory (ABM) that consists of an episodic-like memory (ELM), and a semantic memory (SM), illustrated in Figure 3. The ABM is a PostgreSQL database storing data (essentially from the OPC) each time an action occurs (the episodic memory) but also the knowledge extracted after reasoning about this data (the semantic memory). Thus, the Episodic-Like Memory ELM, is a component of the ABM containing data from the OPC (i.e. current state of the world) before and after each actions done either by the iCub or other agent. Semantic Memory (SM), is a component of the ABM built after reasoning about past experience, based on statistical analysis of data from ELM. The ELM and SM are implemented in postgreSQL. The Autobiographical memory (ABM) is a set of functions that operate on these SQL tables, and interact with the ABM Reasoning module which performs reasoning over past experience and memory consolidation

in order to generate the SM and update the ELM. The Supervisor provides the spoken language interface with the human, and manages high level interaction.

3.1 Episodic-Like Memory

The episode-like memory (ELM) is organized around actions, and the state of the world before and after actions. The SQL data structure of the ELM is illustrated in Figure 4. This action-centered approach is useful in that it helps to solve the problem of how to segment the perceptual stream of events [28]. When the human announces that he will perform an action, a message is sent to the ABM Reasoning, and the current state of the world (a snapshot of the current state of OPC) is stored in the episodic memory in the ELM SQL table, before and after the action occurs. Likewise, the system is informed at the end of each action, and takes an OPC snapshot. With the state of the OPC before and after an action, the robot can extract the pre-condition and effect the for actions [29, 30].

********** Figure 4 about here **********

3.2 Semantic Memory

The semantic memory is derived by ABM Reasoning, from experience encoded in the ELM. ABM Reasoning is coded in C++, and its role is to retrieve the information stored in the ELM and to generalize over this information, in order to extract the pertinent information of each action. The ABM Reasoning thus constructs a Semantic Memory with the pertinent information related to context/spatial/temporal information. The robot will then store its semantic knowledge into the appropriate part of the ABM, i.e. the Semantic Memory. This stored knowledge can be retrieved and reused.

We refer to this as “consolidation” (a dream-like memory consolidation function) [29]. During consolidation, the robot will iterate through all its actions performed in the current session, and will generalize over this data, and consolidate the resulting semantic knowledge in the database. This consolidation is the first level of a system of Retro Reasoning (described in the section 3.3) and is displayed in Figures 6 and 7.

CONSOLIDATION (level 1 reasoning) pseudocode:

For each ACTION **in** the ELM

{

Retrieve the COORDINATES of the object of focus before and after the move.

Calculate the DISPLACEMENT of the object

Populate the corresponding SPATIAL KNOWLEDGE entry and ACTION DEFINITION in the SM.

Calculate the DISPERSION of the displacement to determine if the action is absolute or relative.

If the ACTION is absolute (is location):

Update the LOCATION DEFINITION in the OPC.

}

This pseudo code describes how the encoding of in the ELM of multiple repetitions of actions with spatial parameters like "put north" can be processed to determine that north here refers to a fixed location, which can be learned, and stored in the OPC as a new, learned, named location. This information is also coded in the semantic memory (illustrated in Figure 5) in the spatial data component.

********** Figure 5 about here **********

3.3 Retro Reasoning

Once the robot has extracted these initial concepts, which can be spatial properties related to locations, displacements and actions, it can use this knowledge to construct higher level knowledge. The ABM reasoning will once again iterate through the contents of the ELM, and for actions, will be able to extract knowledge about the pre- and post- condition. For example in the case where a move has a precondition (for example, the moved object must be at location A before going to B), the robot will observe the moves done by the human, and then learn the locations A and B (as outlined in the pseudocode above). Once the robot knows locations A and B, it can then perform retro reasoning, and observe that each time the action: "move-B object" was performed, the object was first at A. This leads to the extraction of the pre-condition: "Object is at A", and similarly for the post condition "Object is at B". Based on this retro reasoning, the system will know that to perform the action "move-B", the condition "Object is in A" is mandatory. Also, the robot will know that the action to perform in order to have "Object is in B" is "move-B object". This information is coded in the context data (before and after fields) of the SM (see Figure 5).

In order to perform retro reasoning, the system requires a copy of the OPC - a mental OPC. In this mental OPC, the robot will simulate the state of the world at the time of the memory, and will analyze it, in the same way it would do it for the real world (i.e. with the new knowledge).

RETRO-REASONING (level 2 reasoning) pseudocode:

For each MEMORY in the ELM (**before** and **after** actions)

{

Re-imagine the SNAPSHOT of the memory in the mental OPC

Re-evaluate the SITUATION with the new spatial knowledge

Integrate the new SPATIAL RELATION found in the ELM

}

********** Figure 6 about here **********

3.4 Level 3 Reasoning

The knowledge that has been acquired through retro reasoning now makes it possible to perform more pertinent reasoning about the conditions that hold before and after actions. This is illustrated in Figure 7B. The ABM Reasoning will again iterate through the ELM (arrow 1) and match this with the knowledge stored in the SM (arrow 2) and will extract new regularities at a higher level. The difference with the first level reasoning is that the ABM Reasoning will now write high level relations in the ELM (arrow 4) such as: "Object is at Location A" that the robot could not have known at the time of the memory (arrow 4), because the location A had not yet been learned. The system can also create higher level knowledge in the SM (arrow 3). For example, in the case of the move of the medium object of the Hanoi Tower, the knowledge will be: "Big object can be at location *From* and location *To*, but the small object can't be at location *From* and location *To*." The function of this level of reasoning is described in the pseudo code below.

LEVEL 3 REASONING (pseudo code)

For each ACTION

{

for STATE before and after the action

{

determine ALL RELATIONS between the object of focus and all objects that hold before and after

calculate PROBABILITY/PERCENTAGE for each relation over all instances in ELM

store appropriately in before and after FIELDS OF CONTEXT in SM

}

}

********** Figure 7 about here **********

Figure 7 then illustrates the system level processing of these different successive levels of representation. Human demonstrations lead to changes in the states of objects as represented in the OPC. These states are recorded in the ELM, associated in time with the named actions, and locations. Level one reasoning detects spatial regularities in terms of the elliptical forms of point cloud distributions of objects in the demonstrated actions and creates this new spatial knowledge.

4 Planning and Goal Directed Reasoning

The previously explained Retro-Reasoning, based on the ELM and SM allows the iCub to obtain information about its known actions, in particular i) pre-conditions, which have to be true if the robot wants to execute the actions and ii) effects, which will be the changes effected in the world with these actions.

This gives the iCub the capacity to know, for a given current state of affairs, what actions are available, and to predict the successive states of the world after several actions. By using these two features, and checking pre-conditions of the next actions against the state of the world attained with the effect of the previous action, the system will be able to reason about a goal and plan successive

action in order to achieve it. This requires the extraction of the acquired knowledge of pre- and post-conditions in a format that can be used for reasoning. This is implemented on the robot by extracting and formatting the rules to be compatible with the standardized planning language PDDL (Planning Domain Definition Language).

********** Figure 9 about here **********

4.1 Planning Domain Definition Language (PDDL) Framework

PDDL is a framework in which the domain of a task can be described (including specification of the "rules" in terms of pre- and post-conditions for actions), and in which a given problem or goal can be specified [19, 20]. This can then be provided as input to a planner, that will attempt to find a sequence of action executions that take the system from the current state, to the specified goal state. Thus, In order to be used, a PDDL planner needs this information, as specified in two different files : a domain, and a problem definition. The domain file contains the set of known actions, including their respective preconditions and effects, whereas in the problem file, we have the current situation description (i.e. the list of all initial conditions) and the desired goal.

Traditionally, these files are hand-coded, with fixed set of actions given to a robot in order to solve a precise kind of problem with variable initial conditions. In our system, these data will be automatically generated in real-time by the iCub, allowing a "developmental" inspired approach based on experience that accumulates and becomes successively refined via the level 1-3 reasoning. This is possible by extracting knowledge from the Semantic Memory (in SQL format) to produce well-formed PDDL domain definition. The problem definition will be made by a direct request to the ABM about the current situation and the goal is defined from interaction with the human. In order to provide a concrete domain in which to pursue this work, we elaborated a simple interaction scenario, illustrated in Figure 8. In this scenario, four spatial locations are learned by the robot, via observation of the human. In addition, the robot learns certain regularities concerning how objects can be moved between these locations. This provides a simple scenario for testing the ability to learn from experience and reason on the acquired knowledge.

********** Figure 8 about here **********

We have extended the architecture from Figure 4, as now illustrated in Figure 9, to allow for the PDDL rule extraction and planning for on-line problem solving. Figure 9 illustrates in more detail the flow of information between the different representations of knowledge, starting in the lowest level “perceptual” representations in the ELM, to the pre-condition, post-condition representations of actions in the PDDL format, appropriate for use with available state of the art reasoning engines [19, 20].The process begins by the expression of the human's desire using speech, indicated by the key-word "want", followed by the goal to reach (1). The Supervisor handles this request and sends it to the ABM module of the iCub (2) which has to solve that problem. The system will then establish the state of the current situation by querying the OPC (3.a, 3.b), and writing it into the problem PDDL file, completed by the human's goal. After that, the system will check the semantic memory to retrieve the contextual knowledge about all the known actions and build the domain PDDL file (5). The AI Planner is then run with these data, to produce a plan (6), made up of the sequence of action which needs to be done in order to achieve the goal from the current situation. This file is parsed and the sequence is sent to the Supervisor (7), which then controls the iCub to execute these moves (8), and thus to achieve the human's goal without any explicit information from him about the "how to".

********** Figure 9 about here **********

5 Experiments

We now demonstrate the operation of the system with two experiments that exercise the ability of the system to extract the structure of knowledge derived from experience, and to reason based on that experience. Both experiments involve interaction tasks that can be organized according to rules or actions that have pre- and post-conditions.

5.1 Experiment 1: Learning Rules About Spatial Movement – Proof of Concept

The goal of the first experiment is to demonstrate that the system is capable of extracting pre-conditions and post-conditions for learned actions, and is then able to use these in the PDDL environment for goal based reasoning in real-time. The robot will learn two types of actions. The first action is to add an object into the interaction space, by putting it on the table. By definition, in terms of our physical constraints, this action can only be performed by the human. The second action is to move an object from one location to another. The two actions will be learned independently. The link between them will be that the precondition of one is the effect of the other. In the first experiment, the initial state will be with the object off the table, as illustrated in Figure 8i, and goal state will be announced to the robot to put the object at location D as illustrated in Figure 8vi. The robot should be able to reason from experience that to put the object at "D", it must be moved from "C", and so on, chaining from the initial state to the final goals state..

********** Figure 10 about here **********

Figure 10 illustrates how the action of moving an object to location B is learned. First, the human has to show the robot how the new actions work by example. He will say to the iCub what he will do (e.g. "I move the circle to B"). The sentence is parsed with the Supervisor, and the recognized action, with the name and the arguments (e.g. "Peter" as agent, "circle" as object, "A" as spatial location), is sent to the ABM, indicating that this action will happen. A snapshot of the OPC is then taken from ABM producing the state of the world before the named action. Again this illustrates how the human interaction allows the system to segment the perceptual flow, here to identify the beginning and ending for actions. Control is returned to the human who can then proceed and execute the action before given a signal to the system ("Done"). This triggers the end of the action, which is written into the ELM, and a second snapshot, this time after the action execution, is then taken. Thus within the ELM there is a specification of the action and its arguments, and snapshots of the state of the world before and after the action. This procedure is repeated several times for the same action (but arguments

could be different) in order to have a set of data where statistical tools could be used for extracting regularities (or "rules"), as described in Section 3.

The characteristic regularity is that actions can be performed with any objects, but there is a *"from-to"* structure that to go to B you must be at A, to C you must be at B, etc. as illustrated in Figure 10. The ABM Reasoning module collects the statistics on the pre- and post-conditions of these movements, and generates a set of entries in the semantic memory "Context" entry, for each type of move, according to its initial and final location.

Once the pre-conditions and effect of actions have been extracted and made explicit in the semantic memory, the system can use them in order to produce the two PDDL files needed for reasoning. The first one, the domain file, is the list of all the known actions, including preconditions and effects, arguments. The ABM Reasoning module begins by writing a "skeleton" of the PDDL, everything which does not change : the domain ("efaa"), the requirements (":strips :typing :equality"), the predicates (isPresent, isAtLoc, Objects and Locations). Then the system iterates through the known actions that are stored in ContextualKnowledge class. For each action the system will write in the files the required components: action, parameters, precondition and effect, which are directly translated from the ContextualKnowledge of the Semantic Memory.

The action name is extracted by combining the verb (e.g. "add", "move") with the none-generalizable arguments (nothing for "add", the location for "move"). That allows for the possibility that actions can have different rules according to the location to where an object is to be moved. The precondition for "move-B" is "isAtLoc obj A", whereas the precondition is "isAtLoc obj C" for "move-D".

Parameters are the arguments over which the action can generalize (e.g. object for "add" and "move"). One can perform these actions with different parameters value, the rules will be the same (for add, the object is not present at first, and is present after, no matter what the object is).

Preconditions are extracted from the ContextualKnowledge class. The system checks for properties which are above a superior threshold for positive conditions and below an inferior threshold for negative conditions, before the action is executed. These properties are the presence of the object (which has to be present for "move" but has not to be for "add") and its location of (for "move-B" the object has to be in location A), as illustrated in Figure 10. The effects or post-conditions are determined in the same way, except that instead of using the regularities before the action is done, the

system computes over the data after the action's execution. Table 1 illustrates the automatically extracted definitions for the ABCD game.

********** Insert Table 1 about here **********

After the creation of the domain file specifying the known actions, the problem file must be produced. The problem is defined by the current state and the goal that is to be achieved. As for the domain extraction, the system begins to write the skeleton of the problem file, with the problem name ("efaa-prob") and the domain name (same as for the domain file, "efaa"). The objects (all the locations and objects known by the iCub) are extracted by a SQL query to the ABM in order to have their name (circle, cross, A, B, …) and their types (object for circle and cross, location A and B). These pairs are added in the "init" section, along with the initial condition.

This PDDL creation is performed when the human asks the robot to reason about a situation, i.e. to attain a particular world state. An OPC snapshot is taken, which correspond to the state of the world before the iCub preforms the reasoning. This snapshot is obtained through an SQL query in order to extract all objects present or absent from the table and if present, their locations. This gives the system the initial situation, which is written in the problem file.

The "goal" part is produced from the human request to the robot to reason about a situation, i.e. to attain a particular world state. Indeed, the human must specify to the robot that he wants something, and enumerate conditions he desires (or does not desires). These are extracted and put inside the goal (e.g. "I want the cross on D" gives "(isAtLoc cross D)"), as illustrated in Table 2.

********** Insert Table 2 about here **********

Both the domain and problem PDDL files are now written. The iCub can run the PDDL planner in order to know what actions he has to do if he wants to execute the human wishes. We use the LPG-td planner [31], with options to find the best of 30 generated solutions, and a computation time limit of 2 seconds. Thus, the system will take a maximum of two seconds or 30 solution files, from lower

quality to better quality before finding the best solution. After execution, the files matching "solutionEFAA_X.SOL" are searched, as X goes from 1 to the maximum number of solutions, to identify the file with the best solution (if there is a solution). The contents of such a file is illustrated in Table 2.

This file is scanned until the actions are found, written between parenthesis. By splitting what is inside according to space, we obtain the name of the action (e.g. "add", "move-A") and the argument on the other part (e.g. "cross" for move). The name is split against but with "-" to have the action verb on one hand, the non-generalizable argument on the other hand. The action is then put together and stored in a YARP bottle, and the next action is parsed until the end of the file. The bottle containing all the actions could now be sent back to the Supervisor Interaction, which will launch the motor command of the iCub to execute them, one by one.

5.2 Experiment 2: The Table of Hanoi

We now consider a more strenuous test of the system. The Table of Hanoi is based on the Tower of Hanoi, adapted to the constraints of the ReacTable. In particular this implies that objects cannot be stacked, but rather they can be placed in zones, following the rules of the Tower of Hanoi, i.e. an object cannot be moved from its current location if there is a smaller object at that same location (because in the ToOH that smaller object would be on top). Also, an object cannot be moved to a location if there is a smaller object at that location. Thus, the goal of the current experiment is to determine if the ABM and domain extraction functions are suitable for learning such rules, and if so, whether the system can learn these rules and then correctly play the TaOH.

In the ABCD experiment, the system was to learn that the constraints on actions involve where objects come from and where they go, but there were no constraints on the objects themselves. The move-A action was demonstrated with different objects, thus there was high variability in the object parameter, and so the object identity was not considered as part of the action, but rather as a free parameter. Thus, the moves could be learned with one set of objects and generalized to another. In the Table of Hanoi experiment, the difference is that, as they have been demonstrated, the actions will be location-generalizable instead of object-generalizable. The Hanoi moves have the same rules from one location to another (between the left, middle and right positions) but they depend on the object involved (small, medium, big), such that small can move to locations with the medium or big object,

medium can move to the big object, and all objects can move to empty locations. Moreover, because the status of the different locations, the origin place (from) and the destination place (to) are particular and have to be managed instead of just working on fixed location.

The human demonstration of these moves (Hanoi-big ?from ?to, Hanoi-medium ?from ?to, Hanoi-small ?from ?to) is done in the same way as for Experiment 1, with only a modification to the lexical entries of the speech recognition grammar (for the new names of objects and locations). It should be noted that, because of the generalization of learning, we need only to perform the moves from "Left" to "Middle", and the robot will be able to generalize to other move locationss. In particular, he has never seen an actual Hanoi game, from the beginning to the end, only a set of illustrative moves. We first teach the robot the locations left, right and middle, by moving each of the three objects three times to each of the locations. This makes 9 moves per location, for a total of 27 moves, which is sufficient to allow the consolidation to extract the location definitions. These locations can then be used to demonstrate the moves that allow the system to learn the rules governing how object positions influence legal moves.

********** Figure 11 about here **********

These moves are illustrated in Figure 11. The three moves executed with the small object indicate that it can move from an occupied or a free position to a free position or an occupied position, thus there are no constraints on where it can come from or go to. This is revealed in the rule that is extracted from the SemanticMemory illustrated in Table 3. For the Medium object, the three demonstations indicate that it cannot move from nor to the same location as the Small. Recall that when calculating the pre-conditions, the system examines all possible relations between objects, and then looks for probabilities that approach 0 (corresponding to a "never" or "not" condition, and probabilities that approach 1 (corresponding to a positive constraint).

For building the domain, the procedure is exactly the same, except the fact that, instead of checking if objects intersect some precise location, the ContextualKnowledge is asked to give information about the "from" and "to", and if the object is on them or not. Nothing changed to write the problem file or launching the PDDL planner.

********** Insert Table 3 about here **********

To specify a problem or goal, the user can set the objects at the desired initial location (on the left location, for eample), so that the system can determine the current state. The user can then state the final state, in terms of the positions of the objects, e.g. that all three objects should be on the right location. This yields the automatic construction of the domain and goal. Table 4 illustrates such a domain and goal specification that was automatically generated. Then, the planner can be run, to generate a solution. The solution is presented in Table 4. Again, the sequence of commands is then automatically transformed into the equivalent commands for the iCub, and the problem is solved, physically.

********** Insert Table 4 about here **********

Figure 14 illustrates the performance of the solution that was generated. Here we see the completion of the embodied reasoning loop: Experience gained by interacting with the human allows the robot to learn the locations, and the rules about different objects and their ability to move to these locations based on the status of other objects in the context (i.e. the rules of the Hanoi game family). Once this knowledge has been extracted it can be automatically formatted into a PDDL description, which can then be executed on standard robust planners. The plan is then automatically transformed into the corresponding sequence of physical actions that can be realized by the iCub, as illustrated in Figure 14.

********** Figure 14 about here **********

6 Discussion and Conclusion

Reasoning requires some form of inference engine, and equally important, a base of structured knowledge from which the system can reason. In the current research, we have conceived and implemented a framework for human-robot interaction, in which, through interaction with the human, the robot acquires experience, and then organizes this experience in order to create a structured

knowledge base from which it can reason. We demonstrate the functioning of the system with two experiments. In the second experiment, from a small set of examples the system learns the rules for moving objects in a Tower of Hanoi – like problem. The system then demonstrates that it can use these rules with a standard AI planner to solve arbitrary problems in the Tower of Hanoi domain. This is of interest, as it illustrates a concrete example where real-world experience, extracted from interaction with a human, can provide a knowledge base upon which an AI system can reason to solve new problems.

When learning new actions, the identification of action parameters is one of the central problems that must be addressed. The difficulty is to determine what are the pertinent aspects of an action, and what can be ignored. For example, Siskind demonstrated how cross situational reasoning can be applied in this context during the acquisition of word meanings [32]. The same kind of statistics can be applied to learning the argument structure of actions [33]. In an effort to determine how to reduce the scope of what should be considered during learning, we previously determined that the focus can be placed on all objects whose state changes as a result of the action [27].

The developing infant faces the same problem, which is, how to know what is the pertinent aspect of a given scene that should be learned. Extensive behavioral studies and observations suggest that in many interactions between adults caregivers and children, the adult creates a very focused context of joint attention with the child in order to supervise in a certain sense what the child will focus on [34]. This motivates us to allow the robot to have knowledge from the user about when demonstrated actions begin and end, particularly when the user is also naming the objects.

In our previous research, the iCub was learning about actions, and we introduced a bias such that actions would generalize over objects. By performing actions such as moving different objects from different starting locations to a fixed target location and calling that “move A to B”, the system detected the variability in the A argument, and thus learned that the move command could take arbitrary arguments for the object. The system thus learned to generalize over objects.

As we have seen, for objects in the Table of Hanoi experiment, the situation is different. Objects are not of a single form of equivalence class. Rather, there are specific rules associated with each object and its movement with respect to the presence and absence of other objects at the source and target destinations. Interestingly, these constraints are coded in the statistical structure of the data in

the ELM, and they are extracted by the multi-level reasoning, to become reflected as pre- and post-conditions of the action representations in the SM.

A principal limitation of the current system is the restrained environment in which it is demonstrated. One can ask whether the current system could generalize to a much more open and high dimensional world, where the focus of interaction would not be so obvious. It has been stated by Levi-Strauss that the objective of man is to understand the world around him [35]. Beneath this objective lies a set of tools for attempting to impose structure on the world. The degrees of freedom for the possible structures that could be imposed on the observables in the world is quite large, and in the absence of constraints, the resulting models or explanations can deviate substantially from the truth, or never converge. This is essentially related to issues of learnability in language, where it has been claimed that the training data that the child is exposed to is so highly under-constrained, that there must be some highly specialized language specific learning capability (reviewed in detail in [36]). However, in the presence of proper constraints, the problem changes. Many typical interactions between infants and caretakers are characterized by behavior that creates joint attention around the object of interest, thus effectively reducing the search space to something very tractable.

This gives us hope that the current approach can scale. By including the human in the learning context, we exploit the notion that the human will perform this search space reduction, by making pertinent demonstrations, and by using language to identify the objects of focus.

7 Acknowledgements

This research was supported by the EFAA Project, funded by FP7-ICT-Challenge 2 Cognitive Systems, Interaction, Robotics Grant Agreement no: 270490.

8 References

[1] F. Hayes-Roth, "Artificial intelligence: What works and what doesn't?," *AI Magazine,* vol. 18, p. 99, 1997.

[2] C. Crangle and P. Suppes, *Language and learning for robots* vol. 41: Center for the Study of Language and Inf, 1994.

[3] S. Lauria, G. Bugmann, T. Kyriacou, and E. Klein, "Mobile robot programming using natural language," *Robotics and Autonomous Systems,* vol. 38, pp. 171-181, 2002.

[4] P. Dominey, A. Mallet, and E. Yoshida, "Real-time cooperative behavior acquisition by a humanoid apprentice," in *International Conference on Humanoid Robotics*, Pittsburg, Pennsylvania, 2007.

[5] P. Dominey, A. Mallet, and E. Yoshida, "Progress in programming the hrp-2 humanoid using spoken language," in *IEEE International Conference on Robotics and Automation*, 2007, pp. 2169-2174.

[6] P. Dominey, A. Mallet, and E. Yoshida, "Real-Time spoken-language programming for cooperative interaction with a humanoid apprentice," *Intl J. Humanoids Robotics,* vol. 6, pp. 147-171, 2009.

[7] F. Doshi and N. Roy, "Spoken language interaction with model uncertainty: an adaptive human–robot interaction system," *Connection Science,* vol. 20, pp. 299-318, 2008.

[8] P. McGuire, J. Fritsch, J. Steil, F. Rothling, G. Fink, S. Wachsmuth, G. Sagerer, and H. Ritter, "Multi-modal human-machine communication for instructing robot grasping tasks," in *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, 2002, pp. 1082-1088.

[9] S. Lallée, S. Lemaignan, A. Lenz, C. Melhuish, L. Natale, S. Skachek, T. van Der Tanz, F. Warneken, and P. Dominey, "Towards a Platform-Independent Cooperative Human-Robot Interaction System: I. Perception," presented at the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Taipei, 2010.

[10] S. Lallée, U. Pattacini, J. Boucher, S. Lemaignan, A. Lenz, C. Melhuish, L. Natale, S. Skachek, K. Hamann, J. Steinwender, E. A. Sisbot, G. Metta, R. Alami, M. Warnier, J. Guitton, F. Warneken, and P. F. Dominey, "Towards a Platform-Independent Cooperative Human-Robot Interaction System: II. Perception, Execution and Imitation of Goal Directed Actions," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, San Francisco, 2011, pp. 2895 - 2902.

[11] S. Lallée, U. Pattacini, S. Lemaignan, A. Lenz, C. Melhuish, L. Natale, S. Skachek, K. Hamann, J. Steinwender, E. A. Sisbot, G. Metta, J. Guitton, R. Alami, M. Warnier, T. Pipe, F. Warneken, and P. Dominey, "Towards a Platform-Independent Cooperative Human-Robot Interaction System: III. An Architecture for Learning and Executing Actions and Shared Plans," *IEEE Transactions on Autonomous Mental Development,* vol. 4, pp. 239-253, 2012.

[12] S. Lallée, F. Warneken, and P. Dominey, "Learning to collaborate by observation," in *Epirob*, Venice, 2009.

[13] M. Petit, S. Lallee, J.-D. Boucher, G. Pointeau, P. Cheminade, D. Ognibene, E. Chinellato, U. Pattacini, Y. Demiris, G. Metta, and P. F. Dominey, "The Coordinating Role of Language in Real-Time Multi-Modal Learning of Cooperative Tasks," *IEEE Transactions on Autonomous Mental Development,* vol. 5, pp. 3-17, 2013.

[14] P. Gorniak and D. Roy, "Grounded semantic composition for visual scenes," *J. Artificial Intelligence Res.,* vol. 21, pp. 429-470, 2004.

[15] D. Roy, "Learning visually grounded words and syntax for a scene description task," *Computer Speech and Language,* vol. 16, pp. 353-385, 2002.

[16] D. Roy and A. Pentland, "Learning words from sights and sounds: A computational model," *Cognitive Sci.,* vol. 26, pp. 113-146, 2002.

[17] S. Carey, *The Origin of Concepts*. Boston: MIT, 2009.

[18] S. Carey and F. Xy, "Infant's knowledge of objects: beyond object files and object tracking," *Cognition,* vol. 80, pp. 179-213, 2001.

[19] M. Helmert, "Concise finite-domain representations for PDDL planning tasks," *Artificial Intelligence,* vol. 173, pp. 503-535, 2009.

[20] D. McDermott, M. Ghallab, A. Howe, C. Knoblock, A. Ram, M. Veloso, D. Weld, and D. Wilkins, "PDDL-the planning domain definition language," 1998.

[21] G. Pointeau, M. Petit, and P. Dominey, "Embodied Simulation Based on Autobiographical Memory," in *Biomimetic and Biohybrid Systems*. vol. 8064, N. Lepora, A. Mura, H. Krapp, P. M. J. Verschure, and T. Prescott, Eds., ed: Springer Berlin Heidelberg, 2013, pp. 240-250.

[22] G. Metta, G. Sandini, D. Vernon, L. Natale, and F. Nori, "The iCub humanoid robot: an open platform for research in embodied cognition," in *PerMIS: Performance Metrics for Intelligent Systems Workshop*, Washington DC, USA, 2008, pp. 19-21.

[23] I. Gori, U. Pattacini, F. Nori, G. Metta, and G. Sandini, "DForC: a Real-Time Method for Reaching, Tracking and Obstacle Avoidance in Humanoid Robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012.

[24] L. Shmuelof and E. Zohary, "Dissociation between ventral and dorsal fMRI activation during object and action recognition," *Neuron,* vol. 47, pp. 457-470, 2005.

[25] S. Sutton, R. Cole, J. Villiers, J. Schalkwyk, P. Vermeulen, M. Macon, Y. Yan, E. Kaiser, B. Rundle, and K. Shobaki, "Universal speech tools: The CSLU toolkit," in *Fifth International Conference on Spoken Language Processing*, 1998.

[26] P. F. Dominey, "Learning grammatical constructions in a miniature language from narrated video events," in *25th Annual Meeting of the Cognitive Science Society*, Boston, 2003.

[27] S. Lallée, C. Madden, M. Hoen, and P. Dominey, "Linking language with embodied teleological representations of action for humanoid cognition," *Frontiers in Neurobotics,* 2010.

[28] J. M. Zacks, N. K. Speer, K. M. Swallow, T. S. Braver, and J. R. Reynolds, "Event perception: a mind-brain perspective," *Psychological bulletin,* vol. 133, p. 273, 2007.

[29] J. D. Payne and L. Nadel, "Sleep, dreams, and memory consolidation: The role of the stress hormone cortisol," *Learning & Memory,* vol. 11, pp. 671-678, 2004.

[30] N. A. Mirza, C. L. Nehaniv, K. Dautenhahn, and R. te Boekhorst, "Developing social action capabilities in a humanoid robot using an interaction history architecture," in *Humanoid Robots, 2008. Humanoids 2008. 8th IEEE-RAS International Conference on*, 2008, pp. 609-616.

[31] A. Gerevini, A. Saetti, I. Serina, and P. Toninelli, "Fast planning in domains with derived predicates: An approach based on rule-action graphs and local search," in *PROCEEDINGS OF THE NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE*, 2005, p. 1157.

[32] J. M. Siskind, "A computational study of cross-situational techniques for learning word-to-meaning mappings," *Cognition,* vol. 61, pp. 39-91, Oct-Nov 1996.

[33] A. Fern, R. Givan, and J. M. Siskind, "Specific-to-general learning for temporal events with application to learning event definitions from video," *Artificial Intelligence Research,* vol. 17, pp. 379-449, 2002.

[34] N. d. V. Rader and P. Zukow-Goldring, "Caregivers' gestures direct infant attention during early word learning: the importance of dynamic synchrony," *Language Sciences,* 2012.

[35] C. Lévi-Strauss, *Myth and meaning*: Psychology Press, 2001.

[36] P. F. Dominey and C. Dodane, "Indeterminacy in language acquisition: the role of child directed speech and joint attention," *Journal of Neurolinguistics,* vol. 17, pp. 121-145, 2004.

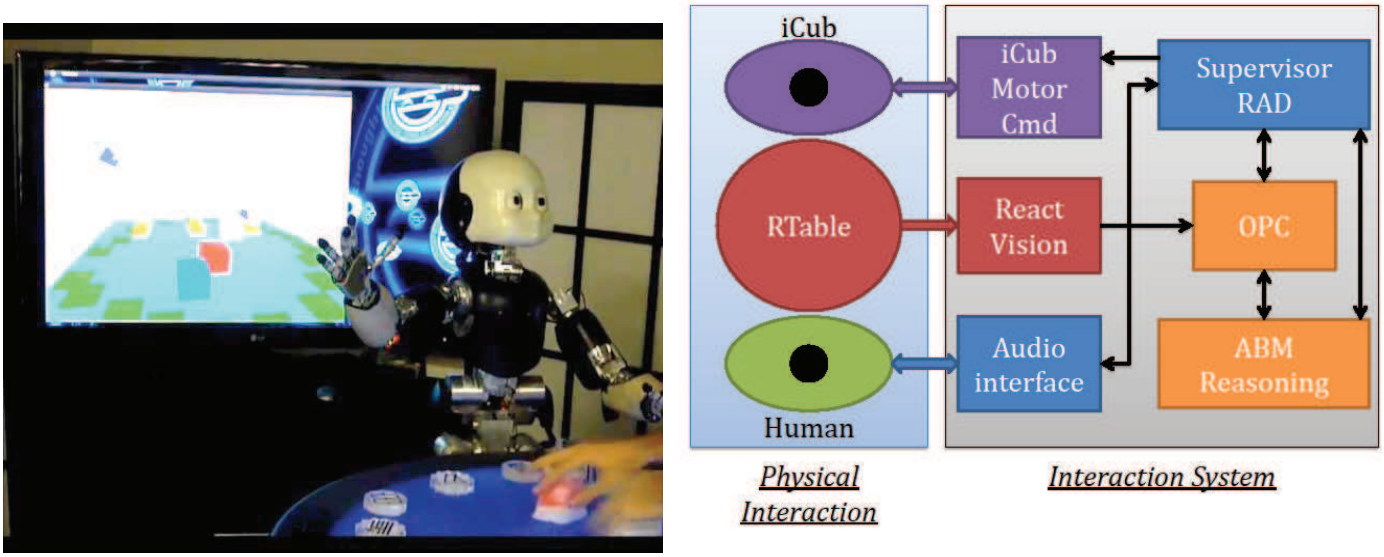


Figure 1. Left. Human-robot physical interaction platform, with the iCub humanoid, and ReacTable interactive table. Screen behind the robot depicts the contents of the OPC (Object Property Collector), reflecting the state of the world on the ReacTable, and the iCub's physical state. Colored objects correspond to objects that are currently perceived on the ReacTable surface. The system can also represent spatial location definitions that have been extracted from the episodic-like memory by consolidation, and are now represented in the Semantic Memory, and have become entities in the OPC. Right. Physical interaction architecture. Human and robot interact by co-manipulating objects on the ReacTable. ReactVision detects objects on the table surface and populates the OPC. The Supervisor manages spoken language interaction. ABM Reasoning manages the autobiographical memory.

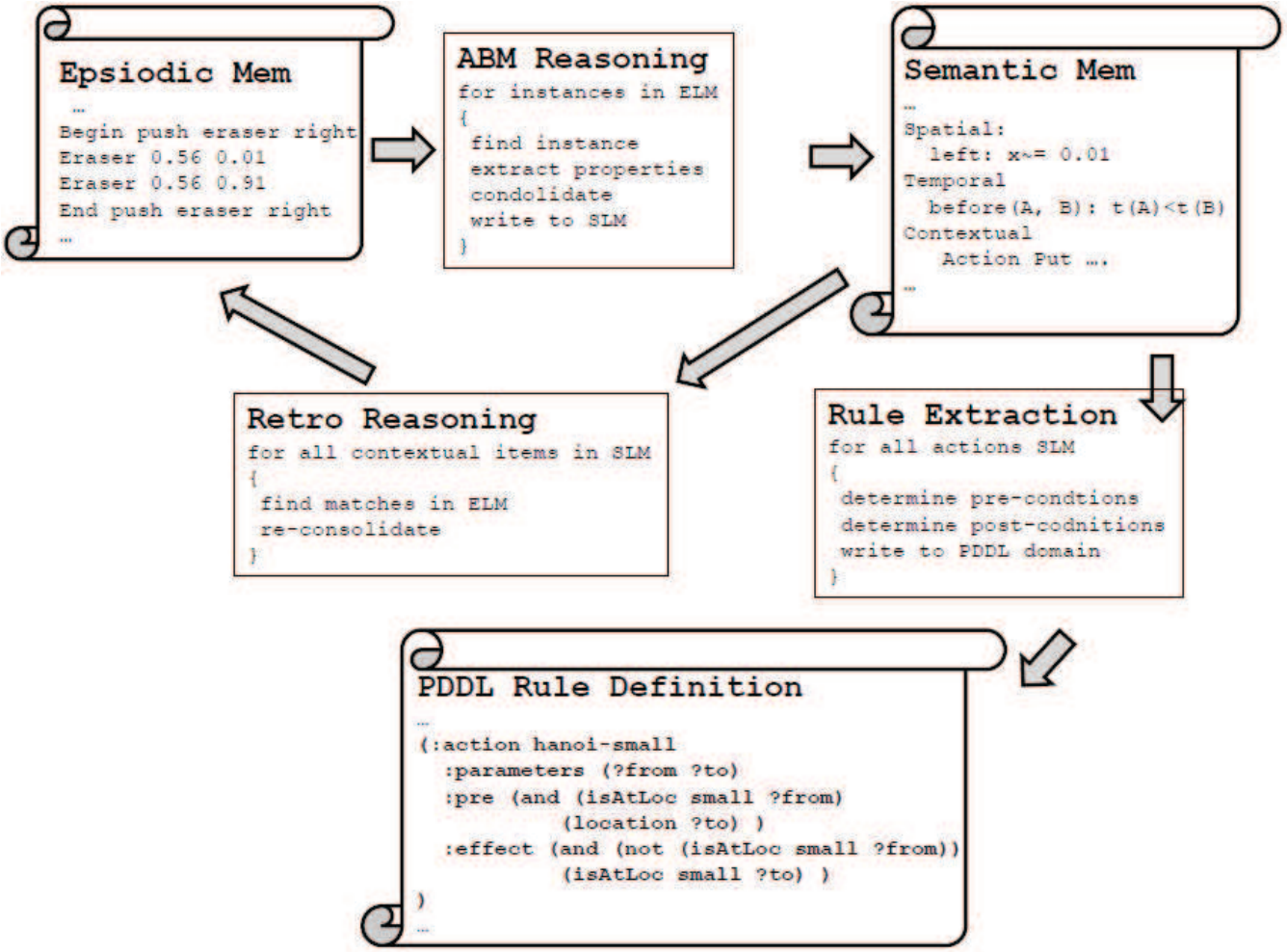


Figure 2. Overview of flow of information in refinement of knowledge. Episodic-like memory (ELM) is a time ordered record of world states (derived from the OPC) before and after each action performed by the human or robot. Through consolidation (by ABM reasoning), regularities about spatial locations, temporal relations, and contextual information about actions are extracted, and encoded in the Semantic memory. By the process of retro-reasoning, this new knowledge is retro-integrated into the ELM (e.g. information about spatial locations that was not known at the time an action was performed). This new information can then propagate and contribute to the contents of semantic memory (e.g. a certain type of action might only be allowed for a specific location, or object). By the process of rule extraction, this information is coded for actions in terms of their pre- and post- conditions, and then automatically reformatted in PDDL, appropriate as input to AI reasoning engines, to allow the system to solve now problems, by reasoning over self-acquired knowledge.

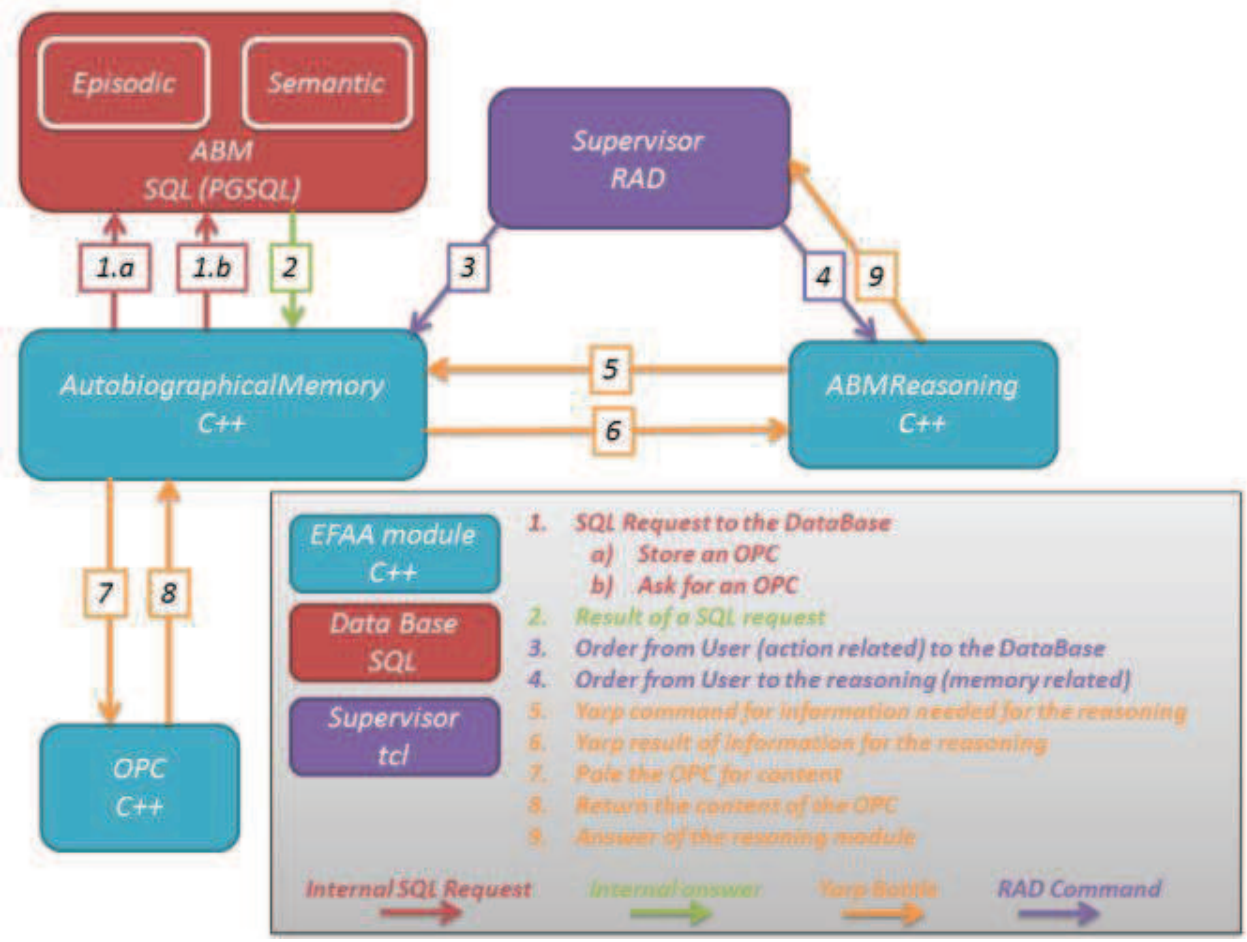


Figure 3. Architecture of the autobiographical memory system. Overview of the memory functioning including the SQL Database, the Supervisor, the ABM Reasoning, and the OPC. 1-2. SQL queries, and replies to ABM are managed by a C++ Autobiographical Memory interface module. 3. User interacts with ABM related to action status, and 4. Memory content. 5-6. ABM reasoning requests and receives content via YARP connections. 7-8. ABM manager requests and receives state data from OPC. 9 Final answer of ABM Reasoning to the supervisor

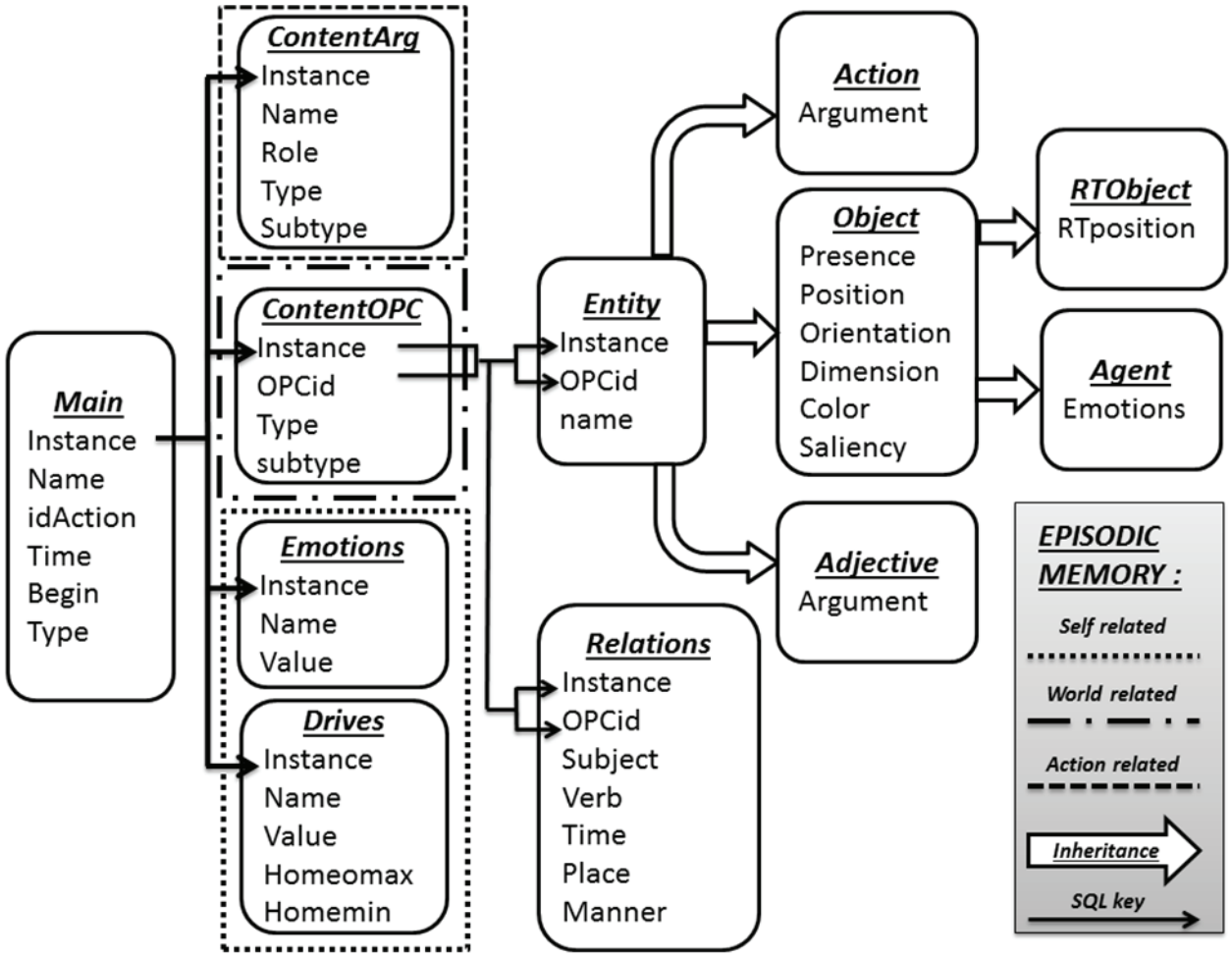


Figure 4. SQL format of the Episodic memory. Architecture of the episodic memory storage in PostgreSQL. The main data type is specified as ContentArg which defines arguments for actions, and ContentOPC which defines entities that are in the OPC. Each interaction has the content of the OPC at a given time (state of the world) but also, information concerning the arguments of the action (who, what, when…). The content of a memory can be divided in 3 sections: self-related, world-related, and action-related

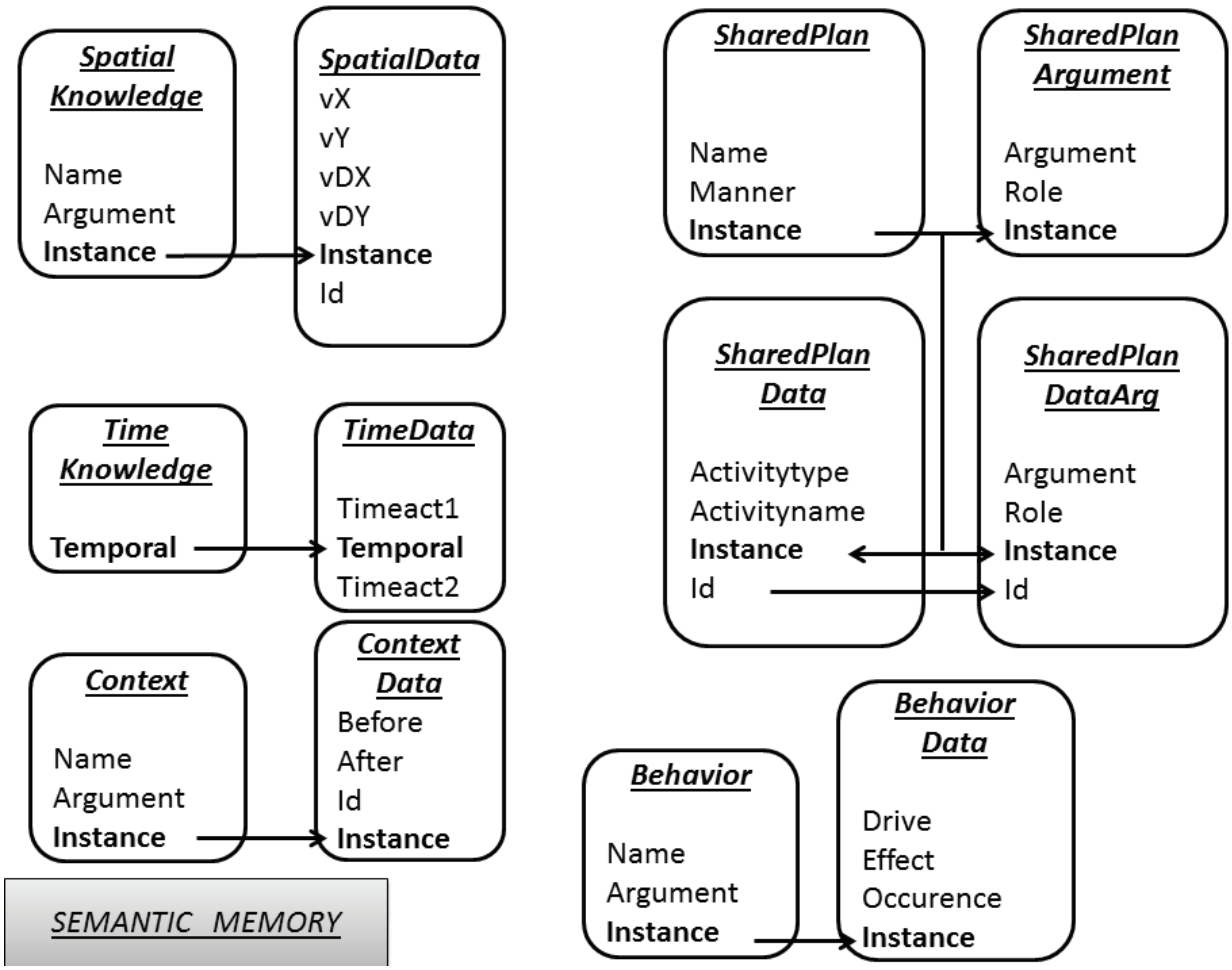


Figure 5. SQL format of the Semantic Memory. Architecture of the semantic memory storage in PostgreSQL. For each type of knowledge, a first table stores the general information concerning the knowledge (name, argument…) while a second table stores the "technical information": the positions of each move in the case of a spatial knowledge, or the time-stamp in the case of a temporal knowledge. For each memory is create an instance (corresponding to a given time). The parameter "instance" allows to get all the information about the state of the world at a given time.

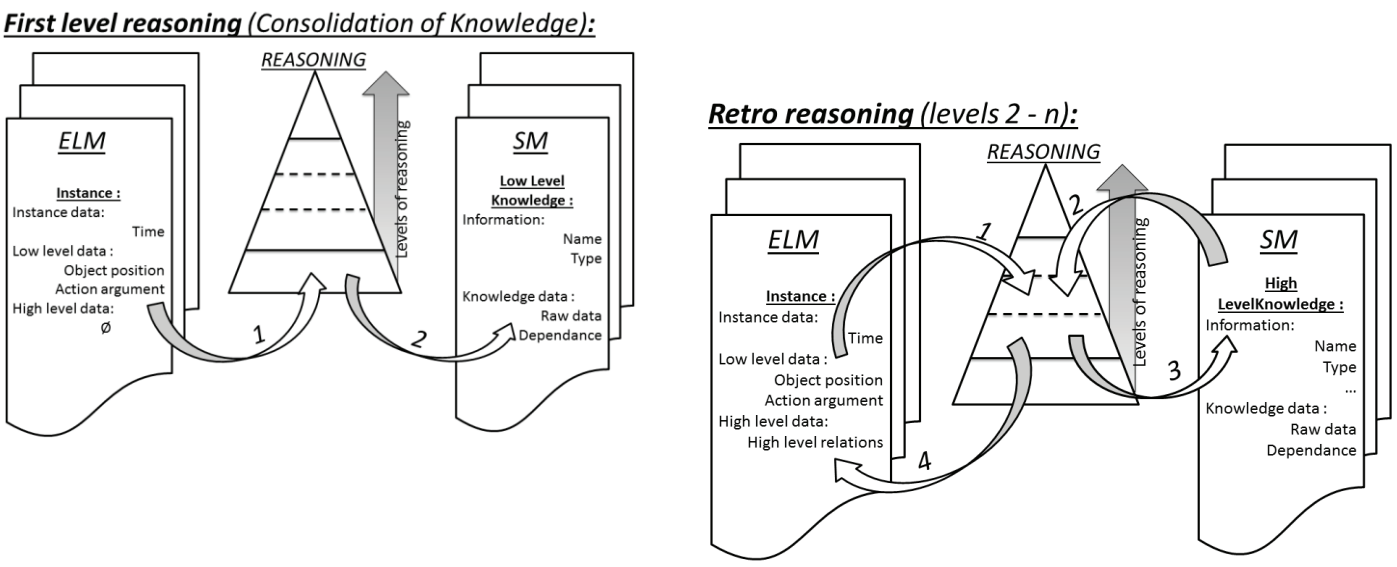


Figure 6: **First level reasoning:** Consolidation of Knowledge Function (first level reasoning). ABM Reasoning gets raw data stored in the ELM, such as: the time of the action, the position of the object, the presence or not of the objects (arrow 1)… The module will then extract the regularities and will build low level knowledge such as the spatial knowledge. For instance the knowledge extracted could be: "location: south_east – raw data of the point cluster constituting the location" or "temporal: before – delay between the 2 actions (here the delay is negative)". The new knowledge will then be stores in the SM (arrow 2). : **Retro reasoning (level 2 - n reasoning).** ABM Reasoning will go through the ELM (arrow 1) and match this with the knowledge stored in the SM (arrow 2) and will extract new regularities at a higher level. The difference with the first level reasoning is that the ABM Reasoning will now write high level relations in the ELM (arrow 4) such as: "Object is at Location – Agent Beliefs are …" that the robot didn't know at the time of the memory (arrow 4). He will also create some high level knowledge in the SM (arrow 4). In the case of the move of the medium object of the Hanoi Tower, the knowledge will be: "Big object can be at location From and location To, and small object can't be at location From and Location To."

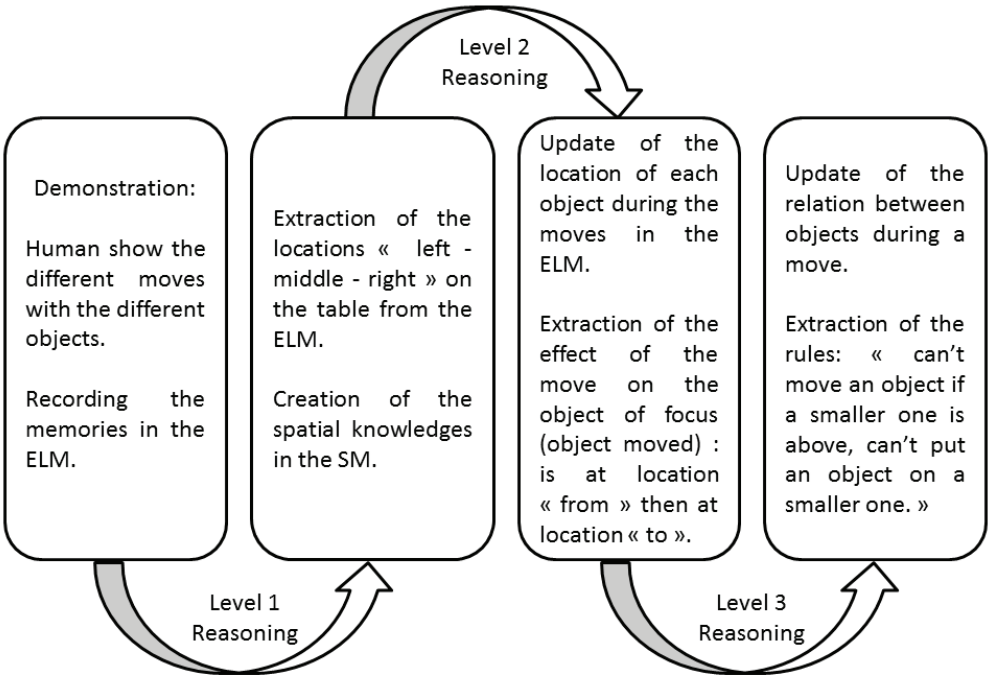


Figure 7: Example of the effect of the different levels of the retro reasoning in the specific case of the Tower of Hanoi.

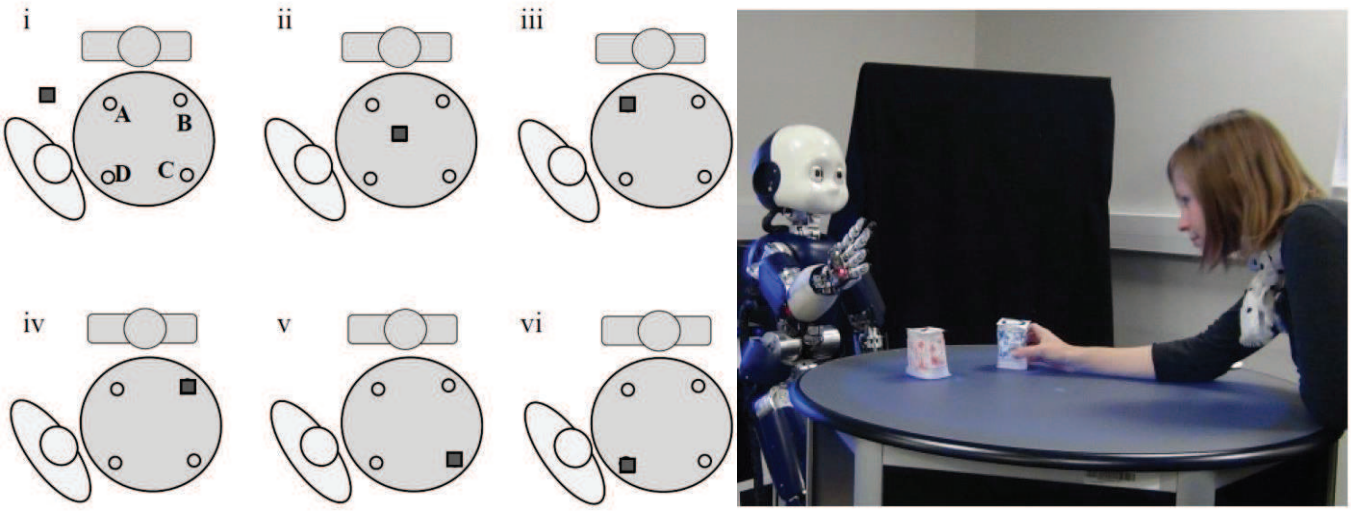


Figure 8. Interaction scenario schema. Four learned locations (on ReacTable (central circle), with human (oval body), iCub (rectangular body) and manipulable object (dark square). The learned locations are labeled A-D. i. Object is off the table. ii. Human has placed the object on the table, in an undefined location. iii. Human or iCub has placed the object location A. iv. Human or iCub has placed the object at B. v-vi. Human has place the object at C, then D, respectively (these locations are out of reach of the iCub).

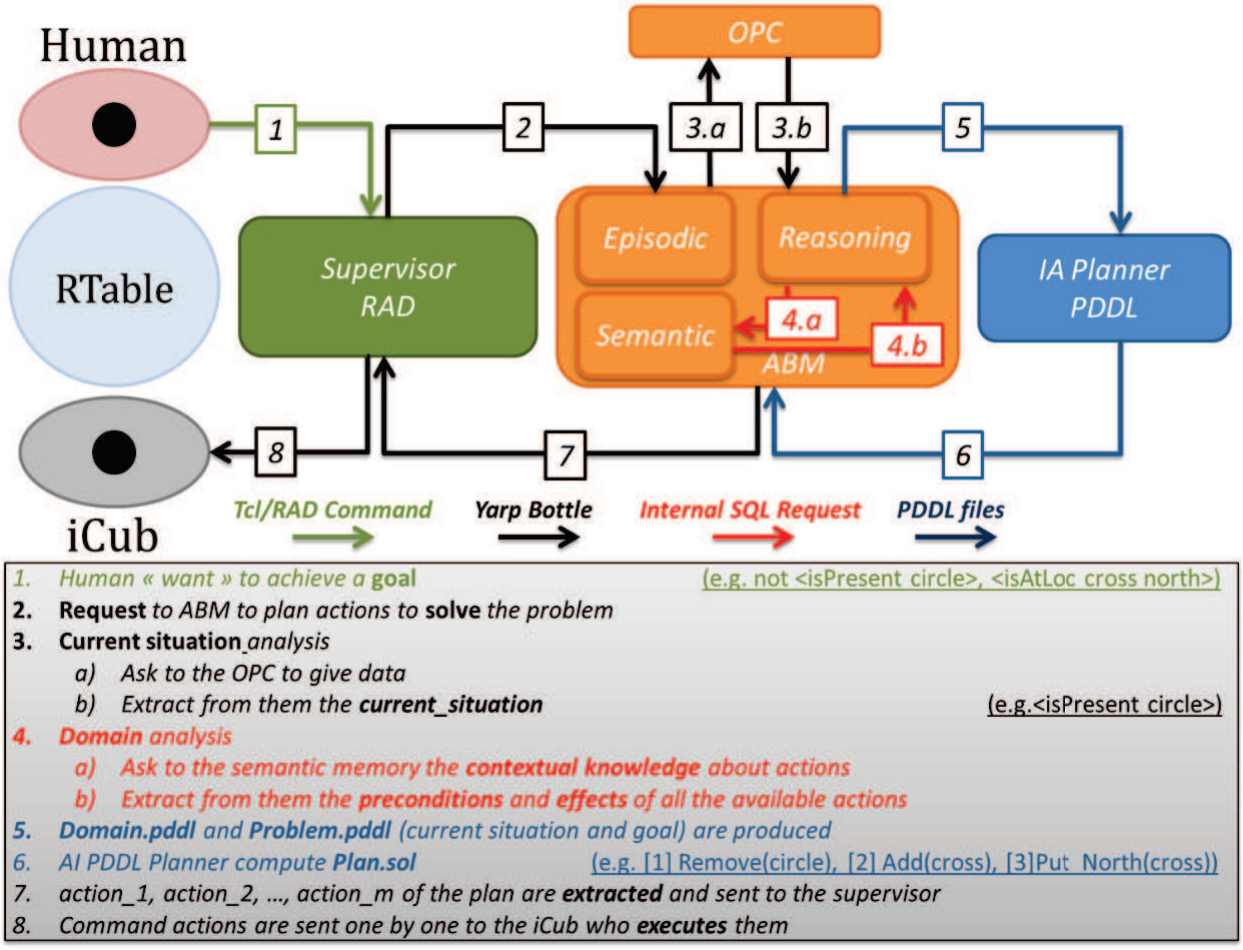
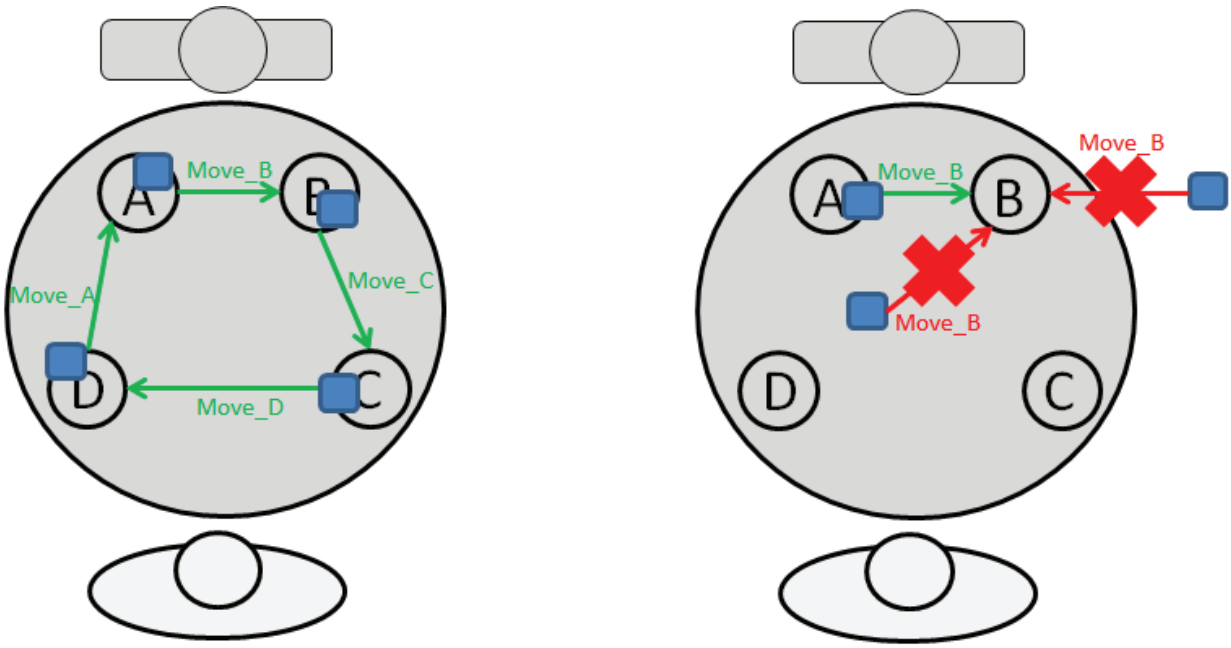


Figure 9. Extended architecture with rule extraction and rule based planning and reasoning.



Example – Move_B
Preconditions : (isPresent obj) (isAtLoc obj A)
Effect : (isPresent obj) (isAtLoc obj B)

Figure 10. Example of definition of different moves in the ABCD experiment. As illustrated, the Move_B action has the precondition that the object must be at location A before Move_B can be executed.

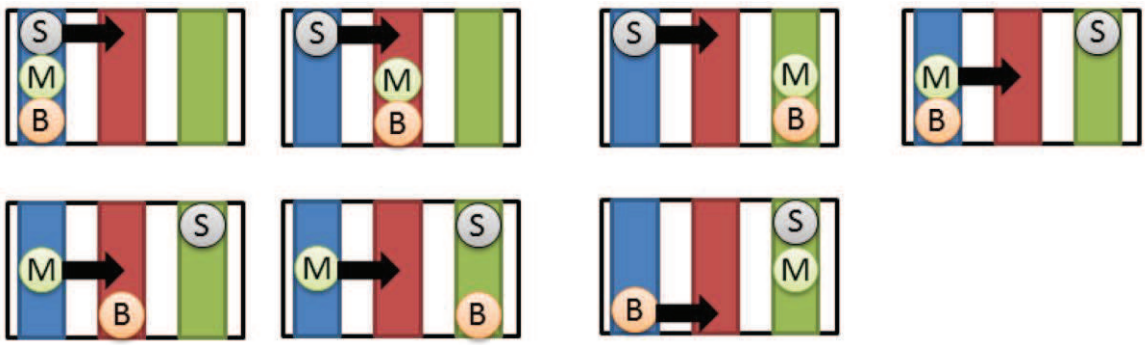


Figure 11. Minimal set of moves to learn the rules for the Table of Hanoi. S,M, B stand for Small, Medium, Big respectively. Colored areas correspond to left, middle and right locations.Three moves are demonstrated with the Small object, 3 for Medium and 1 for Big. See text.

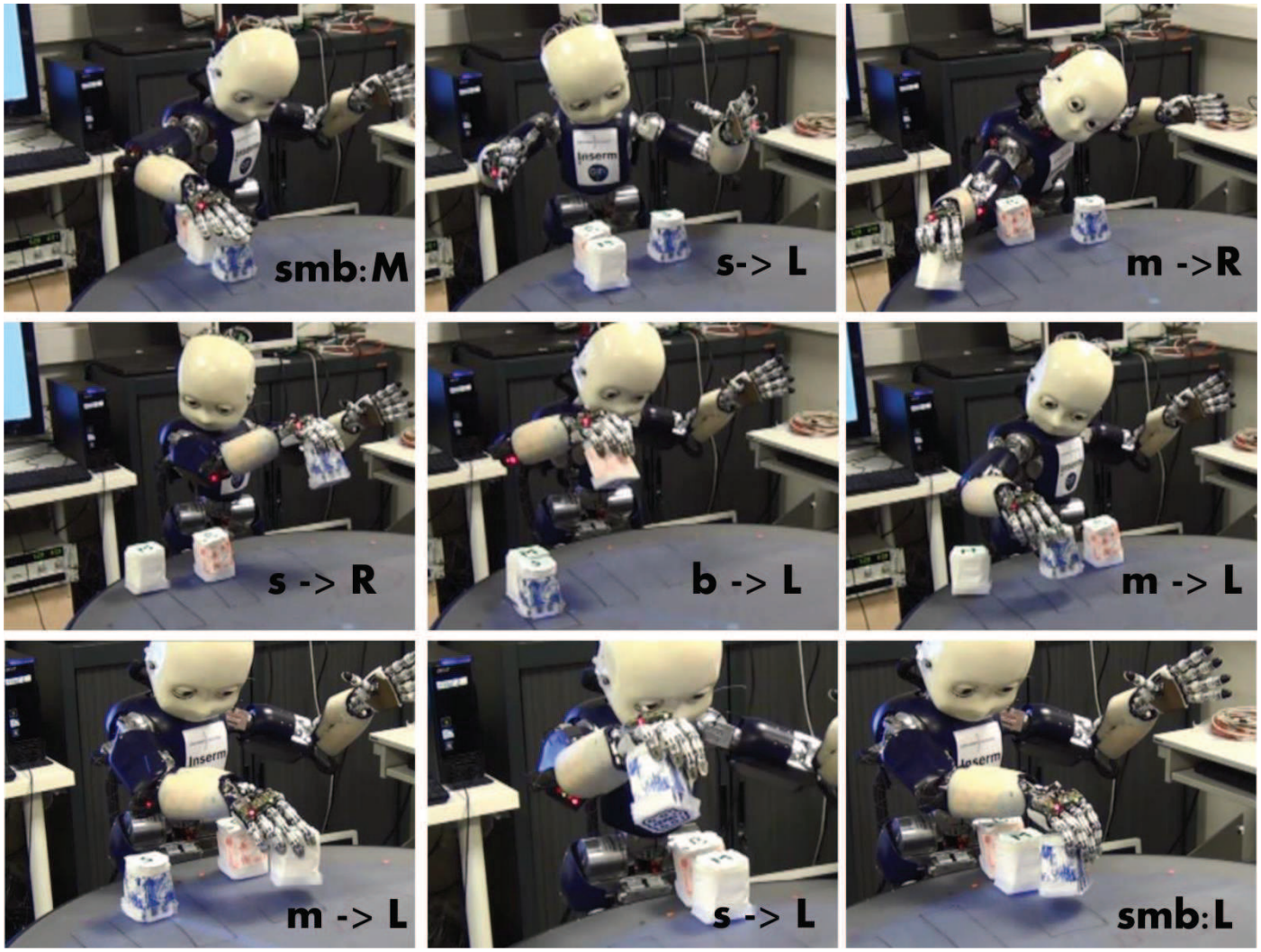


Figure 12. iCub solving the Table of Hanoi. Initial state: small, medium and big objects at Middle position. Goal state: small, medium and big objects at Left position. The problem is solved in 6 moves. (1) small to left, (2) medium to right, (3) small to right, (4) big to left, (5) medium to left, (6) small to left. The task is solved based on learning the rules of the game, without ever seeing a complete solution.

Table 1. Domain knowledge for actions, extracted automatically from the Autobiographical (SLM) memory in PDDL format. Here, for compactness of presentation, we only illustrate the actions related to the ABCD game.

```
;; STRIPS domain automatically generated by ABMReasoning, part of EFAA
(define (domain efaa)
	(:requirements :strips :typing :equality)
	(:predicates
		(isPresent ?obj)
		(isAtLoc ?obj ?loc)
		(Object ?obj)
		(Locations ?loc)
	)
	(:action add
		:parameters (?obj1)
		:precondition (and (not (isPresent ?obj1) ) )
		:effect (and (isPresent ?obj1) )
	)
	(:action remove
		:parameters (?obj1)
		:precondition (and (isPresent ?obj1) )
		:effect (and (not (isPresent ?obj1) ) )
	)
	(:action move-B
		:parameters (?obj1)
		:precondition (and (isPresent ?obj1) (isAtLoc ?obj1 A ))
		:effect (and (isPresent ?obj1) (isAtLoc ?obj1 B) )
	)
	(:action move-C
		:parameters (?obj1)
		:precondition (and (isPresent ?obj1) (isAtLoc ?obj1 B ) )
		:effect (and (isPresent ?obj1) (isAtLoc ?obj1 C) )
	)
	(:action move-D
		:parameters (?obj1)
		:precondition (and (isPresent ?obj1) (isAtLoc ?obj1 C) )
		:effect (and (isPresent ?obj1) (isAtLoc ?obj1 D) )
	)
	(:action move-A
		:parameters (?obj1)
		:precondition (and (isPresent ?obj1) (isAtLoc ?obj1 D ) )
		:effect (and (isPresent ?obj1) (isAtLoc ?obj1 A) )
	)
)
```

Table 2. Specification of the initial state, and the goal, to put the object at location D in the ABCD interaction for Experiment 1.

```
;; STRIPS problem automatically generated by ABMReasoning, part of EFAA
(define (problem efaa-prob)
	(:domain efaa)
	(:objects
		 circle cross eraser
		A B C D
	)	;; end :objects
	(:init
		;;types
		(Object circle) (Object cross) (Object eraser)
		(Locations A) (Locations B) (Locations C) (Locations D)

		;;init-conditions
		(isPresent cross) (isAtLoc cross A)
	)	;; end :init
	(:goal
		(and ( isAtLoc cross D)
		)	;; end and
	)	;; end goal
)	;; end define
```

Solution:

```
; Version LPG-td-1.0
; Seed 52616643
; Command line: lpg-td-1.0 -n 30 -cputime 2 -o domainEFAA.pddl -f problemEFAA.pddl -out solutionEFAA
; Problem problemEFAA.pddl
; Actions having STRIPS duration
; Time 0.05
; Search time 0.00
; Parsing time 0.03
; Mutex time 0.00
; Quality 3


Time 0.05

0:   (MOVE-B CROSS) [1]
1:   (MOVE-C CROSS) [1]
2:   (MOVE-D CROSS) [1]
```

Table 3. Specification of rules for moving the three objects in the Table of Hanoi.

```
;; STRIPS domain automatically generated by ABMReasoning, part of EFAA
(define (domain efaa)
	(:requirements :strips :typing :equality)
	(:types location object)
	(:predicates
		(object ?obj)
		(location ?loc)
		(isAtLoc ?obj ?loc)
	)
	(:action hanoi-small
		:parameters (?from ?to)
		:precondition (and (isAtLoc small ?from) (location ?to) )
		:effect (and (not (isAtLoc small ?from)) (isAtLoc small ?to) )
	)
	(:action hanoi-medium
		:parameters (?from ?to)
		:precondition (and (isAtLoc medium ?from) (not (isAtLoc small ?from))
			(not (isAtLoc small ?to)) (location ?to))
		:effect (and (not (isAtLoc medium ?from)) (isAtLoc medium ?to) )
	)
	(:action hanoi-big
		:parameters (?from ?to)
		:precondition (and (isAtLoc big ?from) (not (isAtLoc small ?from)) (not (isAtLoc small ?to))
			(not (isAtLoc medium ?from)) (not (isAtLoc medium ?to)) (location ?to))
		:effect (and (not (isAtLoc big ?from)) (isAtLoc big ?to) )
	)
)
```

Table 4. Problem and solution to the Table of Hanoi.

```
;; STRIPS problem automatically generated by ABMReasoning, part of EFAA
(define (problem efaa-prob)
        (:domain efaa)
        (:objects
                small medium big
                left middle right
        )       ;; end :objects
        (:init
                ;;types
                (object small) (object medium) (object big)
                (location left) (location right) (location middle)

                ;;init-conditions
                (isAtLoc small left) (isAtLoc medium left) (isAtLoc big left)
        )       ;; end :init
        (:goal
                (and ( isAtLoc big right) ( isAtLoc small right) ( isAtLoc medium right)
                )       ;; end and
        )       ;; end goal
    )   ;; end define
```

Solution:

```
; Version LPG-td-1.0
; Seed 105015930
; Command line: lpg-td-1.0 -o domainEFAA_hanoi.pddl -f problemEFAA_hanoi.pddl -n 30 -cputime 2 -out solu-
tionEFAA_hanoi
; Problem problemEFAA_hanoi.pddl
; Actions having STRIPS duration
; Time 0.03
; Search time 0.01
; Parsing time 0.02
; Mutex time 0.00
; Quality 7


Time 0.03

0:   (HANOI-SMALL LEFT RIGHT) [1]
1:   (HANOI-MEDIUM LEFT MIDDLE) [1]
2:   (HANOI-SMALL RIGHT MIDDLE) [1]
3:   (HANOI-BIG LEFT RIGHT) [1]
4:   (HANOI-SMALL MIDDLE LEFT) [1]
5:   (HANOI-MEDIUM MIDDLE RIGHT) [1]
6:   (HANOI-SMALL LEFT RIGHT) [1]
```

Part III

Discussion

Chapter 6

Discussion and Perspectives

6.1 General Conclusion

The work during this thesis was indeed at the interface between cognitive sciences theories, especially related to children development, and intelligent and social robotics within a developmental framework based on human-robot interactions. The investigation was focused on how to obtain robots with enough advanced cognition capabilities in order to understand, manipulate and adapt in complex human environment. One possible way to achieve that and developed in this work was to implement and exploit unique features from human in their cultural interaction, compared to other species. Indeed, two are particularly important : learning a language to be able to fully make use of cultural transmitted knowledge, and acquiring capacities from experts [Herrmann et al., 2007].

Using a recurrent neural network fed with paired sentence-meaning provided through human-robot interactions, we implemented a system to learn grammatical construction to map and generalize on these structures [Goldberg, 1995]. Thus the robot is exposed to the precise way human are currently speaking to him, and then his language capabilities will be adapted to them, the final users. One particular feature is also that the system is capable of both comprehension (from sentence to meaning) and production (from meaning to sentence) : the robot could then participate in dialogue, and negotiate with the partner in several steps communication.

This capacity to communicate with others is a key feature in cooperative capability in humans : in fact, Tomasello argues that the main function of language is precisely to negotiate during cooperation [Tomasello et al., 2005]. But first, one has to be able to execute actions which will be involved in these collaborative events, and children has several way to learn them if needed, for instance through imitation (when the teacher shows the action), demonstration (when the expert guide and control the learner moves) or through instructions (when the leader details known sub-actions). Language comprehension and production is then required in several scales : i) for the directed instructions itself (as a proper learning modality), ii) to direct the attention of the learner on the precise teaching technique (teacher body for imitation, own body for demonstration, speech for instructions) and iii) to negotiate inside the shared plan, explain it or define who is responsible for what. Using the grammatical construction approach, we have implemented a spoken language interaction system in the robot to manage these shared plan event, allowing the robot to learn a collaborative behavior and to acquire unknown actions involved in it if needed, obtained through a combination of different available modalities.

Thus the robot is able to learn in real-time a shared plan, including acquiring the unknown actions he has to execute within this context, and can coordinate itself with the human in particular in the case of role reversal. However, we do not have yet an intrinsic feature of true collaborative plan : the shared intention. Indeed, the robot could cooperate with the human in order to achieve the plan, but the system has not the actual final goal that both participants (and especially the one who trigger the cooperation) need to share.

That is why we have begin to study more carefully the concept of action representation, especially in a teleological framework, were an action is a mean to achieve a goal when pre-conditions needed for the behavior are encountered [Gergely and Csibra, 2003]. Children can extract many informations from the complex environment using different perceptual capabilities. In order to acquire action concepts, he has to reason about this knowledge and extract among the "noise" what is relevant to the behavior [Csibra and Gergely, 2007]. This mechanism involves a system to store informations and another one to make inference about them. We have then designed in the iCub an autobiographical memory (with both episodic and semantic memory sub-system) to automatically stores information during its life when properties of the world or objects features are extracted as well as symbolic representation coming from spoken description oh the human. Using a description game setup by implementing a semiotic cycle, described in Figure 6.1, we can then ground events description [Steels, 2001, Steels and Baillie, 2003]. A reasoning capability has been designed in order to extract regularities among the raw data thus extracting pre-conditions and effects (which are the action goals), respectively from regularities before and after an action. Formatted in a Planning Domain Definition Language format, these data allows the robot to analyse the current situation and plan the shortest sequence of actions (using a PDDL planner) needed to achieve a goal, potentially shared by the human if he ask the help of the robot. This reasoning capability, based on his own experience, allows the robot to adapt to the current situation, predict outcomes of behaviors and solve unknown problems : he thus gain in flexibility and is more autonomous.

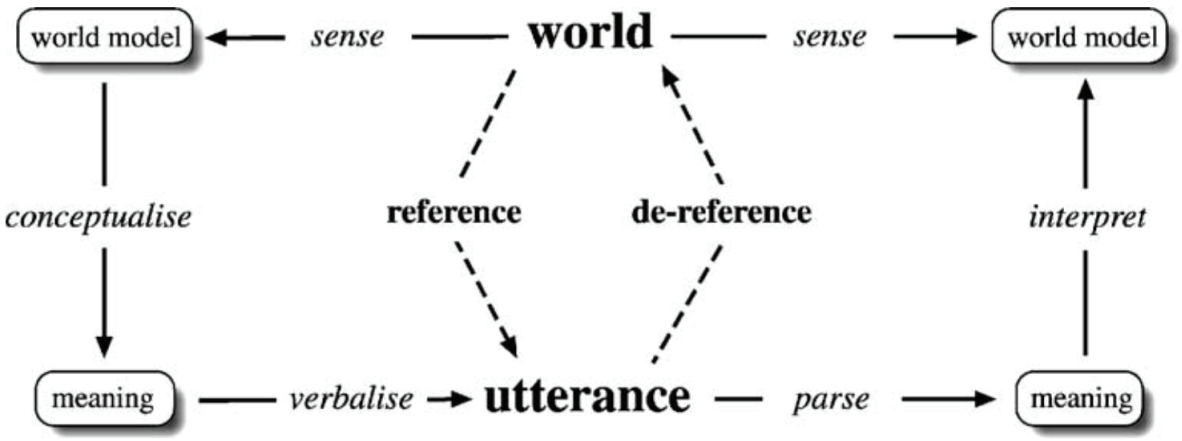


Figure 6.1: *The semiotic cycle for description game interaction, with the processes performed by the speaker to the left, and the ones carried out by the hearer to the right. In particular, when the human is speaking and provides an utterance, the iCub extracts the meaning from it and bring it together with world perception. Sensory data, thus properties of the world, are then linked to semantic concepts (From [Steels and Baillie, 2003]).*

6.2 Discussion

This thesis is largely centered around language, tool of choice to communicate between humans, and is thus an interesting approach for human-robot interactions. This means also that we have access to the symbolic representation carried by the language itself and, according to our child development inspiration, we could use already developed skills from the child when he acquired this features : word segmentation in an utterance, object detection, joint attention, etc ... This has allowed us to focus on the "high-level" aspect of the learning, in particular for conceptualization of actions, and to apply a part of this work an not one but two robots : the Nao and the iCub. But now that we have a working system, capable through human-robot interaction to understand and produce language, to learn and negotiate a shared plan and to reason about actions in order to solve unknown problems, we could look either at the basis or the end of the developmental trajectory. We could then implement i) earlier skills to give the robot some non-mature but already useful capabilities, or ii) advance already matured features in order to be more precise or powerful. As the second point has already been addressed within discussions of the different publications, it is the first point which will be detailed here.

If we look more carefully in language development for children, they go through different stages when they try to acquire their cultural language as shown in Table 6.1. Our recurrent neural network system is defined to use the last step, the verb-general constructions : he defines each roles for open class word and is able to generalize grammatical construction to unknown verb. However, we might encounter a problem of overgeneralization, which is happening also in children between three and four years [Tomasello, 2000] : they apply grammatical constructions without taking care of the verb itself, for instance if it is transitive or intransitive (e.g. "Don’t giggle me").

Approximate age	Experiential scene	Language
9 months	Joint attentional scenes (not symbolized)	—
14 months	Symbolized scenes (undifferentiated symbolization)	Holophrases
18 months	Partitioned scenes (differentiating of event and participant)	Pivot-like constructions
22 months	Syntactic scenes (symbolic marking of participants)	Verb island constructions
36 months	Categorized scenes (generalized symbolic marking of participant roles)	Verb-general constructions

Table 6.1: *Original Caption : Young children's conceptual parsing and categorization of scenes of experience as occasioned by the acquisition of a natural language. (From [Tomasello, 2009]).*

One possibility to fix this issue is then to "go back" a little, and implement the different stages, where semantic is still involved in the process of acquiring the syntax, especially the verb island constructions. For this step, the grammatical constructions are linked to specific and already encountered verbs. Thus, semantic has to be involved in order to constraint the syntax. We could implement a proposed developmental trajectory of Tomasello, shown in Figure 6.2. At first we will have a verb island construction with no generalization, that we will authorized as soon as we have enough different construction. But by keeping the percentage of occurrences for each verb, we could set up an entrenchment system which could generalize only for the same verb subclasses at the end. Thus we abstract the grammatical construction gradually and we constraint along the way.

Lastly, we have presented a long-term memory framework which could store raw data from interaction and language in episodic events that an inference reasoning mechanism could analyze and extract regularities, leading to semantic concepts. However, we have been interesting in the concept of actions itself, so taken independently from the other. This allowed us to reason about them and produce new and efficient sequences of actions depending on the current situation and the goal to achieve. But what about sequences of actions which are inherently linked together to form habits (e.g. always pointing to an object before moving it), social convention (e.g. look at the partner, saying "thank you" then smiling when we received something from him) or more generaly in any turn-taking activity (e.g. peek-a-boo game). To keep together these actions, an implementation of a short-term memory could be a solution. Broz and colleagues have indeed recently produce a system called Extended Interaction History Architecture (EIHA), containing a short-term memory which can handle and sequences of simple actions, associating with

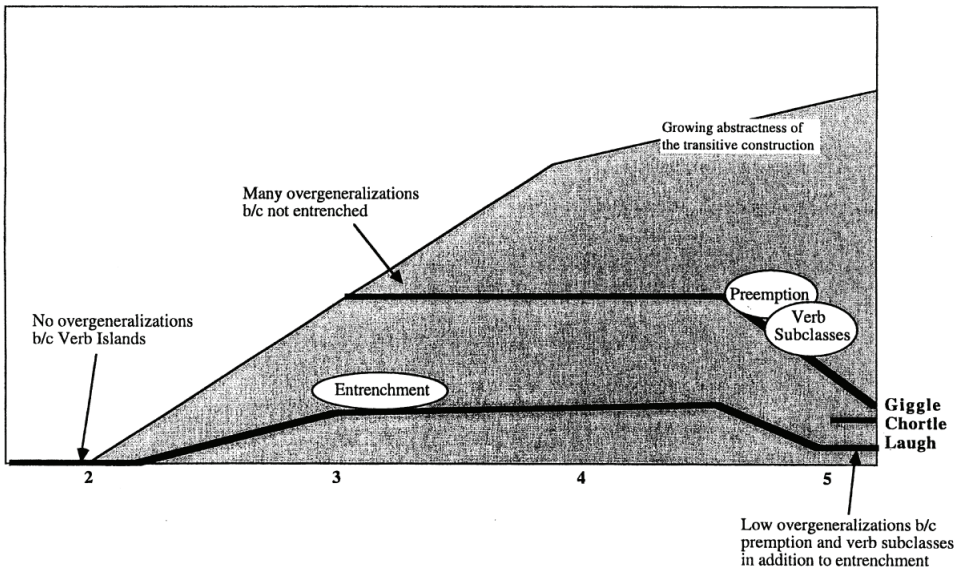


Figure 6.2: *Original Caption : Shaded area depicts growing abstractness of the transitive construction. Other specifications designate constraints on the tendency to overgeneralize inappropriate verbs to this construction. (From [Tomasello, 2000]).*

conditions, rewards, etc... [Broz et al., 2012]. They achieved to learn a pick-a-boo behavior ("hide-face" then "home-position") and a drumming activity ("start-drum", several "drum-hit", "home-position") : this last behavior, more complex, is not learned by the system if the short-term memory capability is removed, thus this feature is a key component in EIHA in order to learn complex turn-taking interactions.

Eventually, we have shown how a robot can learn complex actions and shared plan using its language understanding and production acquired skills. But children can learn simple actions before they are able to produce complete and correct sentences. Our robot has first to learn how to speak and then could learn through spoken interactions with the humans. If we want for the robot to be effective earlier, we will need to address the learning problem without language capability. Recent work has been made to teach a robot without any specific reward and no linguistic feedback nor cues : for instance the interaction rhythm and synchrony could be used as natural reinforcement signals for learning move through mirroring [Hiolle et al., 2010, Prepin and Gaussier, 2010, Andry et al., 2011]. Indeed, they have shown that synchrony is naturally attained when the human is satisfied by the actions of the robot providing then a positive feedback. The interesting part is that Hiolle specifically give instructions to the naive human subjects, where they are told to teach the robot as if he was a 6 to 15 month old infant and that he could process speech and facial emotions (which in fact was not the case). Indeed, the subjects were acting more naturally and thus provided coherent rhythm which could be exploit by the robot, along with voice tons or facial expression, not used but still present in human behavior. The belief of the naive human subject in different communicative skills of the robot is thus very important : social robots are not designed to be used by expert users exclusively, we need to think about naive subject and how they will interact with the embodied cognitive system. We need to encourage them, and adding robot expressiveness or keeping them in the dark on the actual capacity to process speech or facial emotions lead them to longer

and better interactions [Hiolle et al., 2010].

In fact, this precise point is also related to the other aspect of this thesis : each of them was using human-robot interactions with the intention to be used by naive subjects. You want to predict how humans will act with a robot, how they will respond to different actions and trying to extract their feedback in order to build a coherent system. And thus, several and powerful algorithms, concepts and apparatus could be used. But at the end, if the human does not act as expected, does not answer to the robot, look at the experimenter instead of in the eyes of the iCub, you will just have noise data and nothing useful to extract from them. As roboticist, we are the first one to use and test our interaction system, and we also know how it is working, what is expecting, etc... Taking time to implement "toy" or "esthetic" functions, like blinking, respiratory moves, waiting actions (e.g. looking back and forth to the human and a focus object) in order to give a feeling of "life" in the robot, could change completely the naive subject behaviors and engagement and thus the efficiency of the system. Even if you can (and arguably have to) used intensively the human in developmental approach for social robotics, at the end the human is not forced to interact with him and teach something. We then need to be sure that we could encourage such episodes and that they are pleasant, or at least interesting for the human, and seen as a distractful game to avoid frustration or boredness. Eventually at the end, we has to not forget that even if humans can help the robot to take their first steps, it is the robot which has to entertain or give assistance to the human, and not the other way around.

Part IV

Appendix

A. List of Figures

B. List of Tables

Bibliography

al Jazari, Ibn al-Razzaz. *The Book of Knowledge of Ingenious Mechanical Devices*. Dordrecht Publishing Company, translated by Donald Routledge Hill in 1974, 1206.

Andry, Pierre, Blanchard, Arnaud, and Gaussier, Philippe. Using the rhythm of nonverbal human–robot interaction as a signal for learning. *Autonomous Mental Development, IEEE Transactions on*, 3(1):30–42, 2011.

Asada, Minoru, MacDorman, Karl F, Ishiguro, Hiroshi, and Kuniyoshi, Yasuo. Cognitive developmental robotics as a new paradigm for the design of humanoid robots. *Robotics and Autonomous Systems*, 37(2):185–193, 2001.

Ashley, Jennifer and Tomasello, Michael. Cooperative problem-solving and teaching in preschoolers. *Social Development*, 7(2):143–163, 1998.

Bakeman, Roger and Adamson, Lauren B. Coordinating attention to people and objects in mother-infant and peer-infant interaction. *Child development*, pages 1278–1289, 1984.

Baron-Cohen, Simon. *Mindblindness: An essay on autism and theory of mind*. MIT press, 1997.

Bates, Elizabeth, Wulfeck, Beverly, and MacWhinney, Brian. Cross-linguistic research in aphasia: An overview. *Brain and Language*, 41(2):123–148, 1991.

Bauer, Patricia J, Wenner, Jennifer A, Dropik, Patricia L, Wewerka, Sandi S, and Howe, Mark L. Parameters of remembering and forgetting in the transition from infancy to early childhood. *Monographs of the Society for Research in Child Development*, pages i–213, 2000.

Bedini, Silvio A. The role of automata in the history of technology. *Technology and Culture*, 5(1):24–42, 1964.

Beer, Randall D. A dynamical systems perspective on agent-environment interaction. *Artificial intelligence*, 72(1):173–215, 1995.

Bencini, Giulia ML and Goldberg, Adele E. The contribution of argument structure constructions to sentence meaning. *Journal of Memory and Language*, 43(4):640–651, 2000.

Bonawitz, Elizabeth Baraff, Ferranti, Darlene, Saxe, Rebecca, Gopnik, Alison, Meltzoff, Andrew N, Woodward, James, and Schulz, Laura E. Just do it? investigating the gap between prediction and action in toddlers' causal inferences. *Cognition*, 115(1):104–117, 2010.

Braitenberg, Valentino. *Vehicles: Experiments in synthetic psychology*. MIT press, 1986.

Bratman, Michael E. Shared cooperative activity. *The philosophical review*, 101(2):327–341, 1992.

Brinck, Ingar and Gärdenfors, Peter. Co–operation and communication in apes and humans. *Mind & Language*, 18(5):484–501, 2003.

Broen, Patricia Ann. *The Verbal Environment of the Language-Learning Child. ASHA Monographs, No. 17.* ERIC, 1972.

Brooks, Rechele and Meltzoff, Andrew N. The development of gaze following and its relation to language. *Developmental science*, 8(6):535–543, 2005.

Brooks, Rodney. A robust layered control system for a mobile robot. *Robotics and Automation, IEEE Journal of*, 2(1):14–23, 1986.

Brooks, Rodney A. Intelligence without representation. *Artificial intelligence*, 47(1):139–159, 1991.

Brooks, Rodney A and Stein, Lynn Andrea. Building brains for bodies. *Autonomous Robots*, 1(1):7–25, 1994.

Brown, Roger and Hanlon, Camille. Derivational complexity and order of acquisition in child speech. *Cognition and the development of language. New York: Wiley*, 8, 1970.

Brownell, Celia A and Carriger, Michael Sean. Changes in cooperation and self-other differentiation during the second year. *Child development*, 61(4):1164–1174, 1990.

Brownell, Celia A, Ramani, Geetha B, and Zerwas, Stephanie. Becoming a social partner with peers: Cooperation and social understanding in one-and two-year-olds. *Child Development*, 77(4):803–821, 2006.

Broz, Frank, Nehaniv, Chrystopher L, Kose-Bagci, Hatice, and Dautenhahn, Kerstin. Interaction histories and short term memory: Enactive development of turn-taking behaviors in a childlike humanoid robot. *arXiv preprint arXiv:1202.5600*, 2012.

Cangelosi, Angelo, Metta, Giorgio, Sagerer, Gerhard, Nolfi, Stefano, Nehaniv, Chrystopher, Fischer, Kerstin, Tani, Jun, Belpaeme, Tony, Sandini, Giulio, Nori, Francesco, et al. Integration of action and language knowledge: A roadmap for developmental robotics. *Autonomous Mental Development, IEEE Transactions on*, 2(3):167–195, 2010.

Carpenter, Malinda, Nagell, Katherine, Tomasello, Michael, Butterworth, George, and Moore, Chris. Social cognition, joint attention, and communicative competence from 9 to 15 months of age. *Monographs of the society for research in child development*, pages i–174, 1998.

Carpenter, Malinda, Tomasello, Michael, and Striano, Tricia. Role reversal imitation and language in typically developing infants and children with autism. *Infancy*, 8(3): 253–278, 2005.

Chapuis, Alfred and Droz, Edmond. *Automata: A historical and technological study.* Éditions du Griffon, 1958.

Cheng, Patricia W. From covariation to causation: A causal power theory. *Psychological review*, 104(2):367, 1997.

Chomsky, Noam. On certain formal properties of grammars. *Information and control*, 2 (2):137–167, 1959.

Chomsky, Noam. *Aspects of the Theory of Syntax*, volume 11. The MIT press, 1965.

Chomsky, Noem. *Syntactic structures.* Mouton, 1957.

Cohen, Gillian and Conway, Martin A. *Memory in the real world.* Psychology Press, 2007.

Conway, Martin A and Pleydell-Pearce, Christopher W. The construction of autobiographical memories in the self-memory system. *Psychological review*, 107(2):261, 2000.

Csibra, Gergely and Gergely, Gyorgy. 'obsessed with goals': Functions and mechanisms of teleological interpretation of actions in humans. *Acta psychologica*, 124(1):60–78, 2007.

Dautenhahn, Kerstin. Getting to know each other—artificial social intelligence for autonomous robots. *Robotics and autonomous systems*, 16(2):333–356, 1995.

DeCasper, Anthony J and Spence, Melanie J. Prenatal maternal speech influences newborns' perception of speech sounds. *Infant behavior and Development*, 9(2):133–150, 1986.

Dominey, Peter F and Dodane, Christelle. Indeterminacy in language acquisition: the role of child directed speech and joint attention. *Journal of Neurolinguistics*, 17(2):121–145, 2004.

Eckerman, Carol O and Didow, Sharon M. Nonverbal imitation and toddlers' mastery of verbal means of achieving coordinated action. *Developmental Psychology*, 32(1):141, 1996.

Eckerman, Carol O and Peterman, Karen. *Chapter twelve peers and infant social/communicative development*, volume 326. Blackwell handbook of infant development, 2001.

Edmonds, Bruce. Capturing social embeddedness: a constructivist approach. *Adaptive Behavior*, 7(3-4):323–347, 1999.

Engelberger, Joseph F. *Robotics in Practice: Management and Applications of Industrial Robots.* Kogan Page Ltd., London, 1980.

Fagan, Joseph F. Infants' delayed recognition memory and forgetting. *Journal of Experimental Child Psychology*, 16(3):424–450, 1973.

Farmer, Henry George. *The Organ of the Ancients, From Eastern Sources (Hebrew, Syriac and Arabic).* 1st edition. W. Reeves, New-York, 1931.

Fernald, Anne. Four-month-old infants prefer to listen to motherese. *Infant behavior and development*, 8(2):181–195, 1985.

Fernald, Anne. Intonation and communicative intent in mothers' speech to infants: Is the melody the message? *Child development*, pages 1497–1510, 1989.

Fernald, Anne and Mazzie, Claudia. Prosody and focus in speech to infants and adults. *Developmental psychology*, 27(2):209, 1991.

Fisher, Cynthia and Tokura, Hisayo. Acoustic cues to grammatical structure in infant-directed speech: Cross-linguistic evidence. *Child Development*, 67(6):3192–3218, 1996.

Fitzpatrick, Paul and Metta, Giorgio. Grounding vision through experimental manipulation. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 361(1811):2165–2185, 2003.

Fivush, Robyn. The development of autobiographical memory. *Annual review of psychology*, 62:559–582, 2011.

Fowler, Charles B. The Museum of Music: A History of Mechanical Instruments. *Music Educators Journal*, 54(2):45–49, 1967.

Gallese, Vittorio and Lakoff, George. The brain's concepts: The role of the sensory-motor system in conceptual knowledge. *Cognitive neuropsychology*, 22(3-4):455–479, 2005.

Gallup, Gordon G. Chimpanzees: self-recognition. *Science*, 167(3914):86–87, 1970.

Galluzzi, Paolo. *The career of a technologist, Leonardo da Vinci, Engineer and Architect*. Montreal: Montreal Museum of Fine Arts, 1987.

Garrard, Mary D. *Brunelleschi's Egg: Nature, Art, and Gender in Renaissance Italy*. Berkely/Los Angeles/London: University of California Press, 1987.

Gergely, György and Csibra, Gergely. Teleological reasoning in infancy: The naıve theory of rational action. *Trends in cognitive sciences*, 7(7):287–292, 2003.

Gergely, Gyorgy and Csibra, Gergely. The social construction of the cultural mind: Imitative learning as a mechanism of human pedagogy. *Interaction Studies*, 6(3):463–481, 2005.

Gergely, György, Nádasdy, Zoltán, Csibra, Gergely, and Bíró, Szilvia. Taking the intentional stance at 12 months of age. *Cognition*, 56(2):165–193, 1995.

Gergely, György, Bekkering, Harold, and Király, Ildikó. Developmental psychology: rational imitation in preverbal infants. *Nature*, 415(6873):755–755, 2002.

Gillette, Jane, Gleitman, Henry, Gleitman, Lila, and Lederer, Anne. Human simulations of vocabulary learning. *Cognition*, 73(2):135–176, 1999.

Gold, E Mark. Language identification in the limit. *Information and control*, 10(5): 447–474, 1967.

Goldberg, Adele E. *Constructions: A construction grammar approach to argument structure*. University of Chicago Press, 1995.

Golinkoff, Roberta Michnick, Jacquet, Roberta Church, Hirsh-Pasek, Kathy, and Nandakumar, Ratna. Lexical principles may underlie the learning of verbs. *Child development*, 67(6):3101–3119, 1996.

Halle, Morris. Phonology in generative grammar. *Word*, 18(1/2):54–72, 1962.

Harnad, Stevan. The symbol grounding problem. *Physica D: Nonlinear Phenomena*, 42 (1):335–346, 1990.

Hartup, Willard W. Social relationships and their developmental significance. *American psychologist*, 44(2):120, 1989.

Hayes-Roth, Frederick. Artificial intelligence: What works and what doesn't? *AI Magazine*, 18(2):99, 1997.

Henderson, Harry. *Modern Robotics: Building Versatile Machines*. Infobase Publishing, 2006.

Hero. *The Pneumatics of Hero of Alexandria from the original Greek*. translated for and edited by Bennet Woodcroft in 1851. London, Taylor, Walton and Maberly, c. A.D. 85.

Herrmann, Esther, Call, Josep, Hernández-Lloreda, María Victoria, Hare, Brian, and Tomasello, Michael. Humans have evolved specialized skills of social cognition: the cultural intelligence hypothesis. *science*, 317(5843):1360–1366, 2007.

Hill, Donal R. *Studies in Medieval Islamic Technology: From Philo to al-Jazari - From Alexandria to Diyar Bakr*. Edited by David A. King (Variorium Collected Studies Series, 555). Aldershot, Eng./Brookfield, Vt. : Ashgate, 1998.

Hiolle, Antoine, Cañamero, Lola, Andry, Pierre, Blanchard, Arnaud, and Gaussier, Philippe. Using the interaction rhythm as a natural reinforcement signal for social robots: a matter of belief. In *Social Robotics*, pages 81–89. Springer, 2010.

Holland, Owen. Exploration and high adventure: the legacy of grey walter. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 361(1811):2085–2121, 2003.

Hood, Bruce M, Willen, J Douglas, and Driver, Jon. Adult's eyes trigger shifts of visual attention in human infants. *Psychological Science*, 9(2):131–134, 1998.

Horáková, Jana and Kelemen, Jozef. Robots between Fictions and Facts. In *Intl Symposium on Computational Intelligence and Informatics*, pages 21–39, 2006.

Kaplan, Frederic, Oudeyer, Pierre-Yves, and Bergen, Benjamin. Computational models in the debate over language learnability. *infant and child development*, 17(1):55–80, 2008.

King, Barbara J. Social information transfer in monkeys, apes, and hominids. *American Journal of Physical Anthropology*, 34(S13):97–115, 1991.

Koetsier, Teun. On the prehistory of programmable machines: musical automata, looms, calculators. *Mechanism and Machine Theory*, 36(5):586–603, 2001.

Kruger, Ann C and Tomasello, Michael. Cultural learning and learning culture. *The handbook of education and human development: New models of learning, teaching and schooling*, pages 369–387, 1996.

Landes, Joan B. Vaucanson's Automata as Devices of Enlightenment. *Sjuttonhundratal*, 8:50–59, 2012.

Leong, Deborah J. Scaffolding emergent writing in the zone of proximal development. *Literacy*, 3(2):1, 1998.

Lindblom, Jessica and Ziemke, Tom. Social situatedness of natural and artificial intelligence: Vygotsky and beyond. *Adaptive Behavior*, 11(2):79–96, 2003.

Lungarella, Max, Metta, Giorgio, Pfeifer, Rolf, and Sandini, Giulio. Developmental robotics: a survey. *Connection Science*, 15(4):151–190, 2003.

MacWhinney, Brian. A multiple process solution to the logical problem of language acquisition. *Journal of child language*, 31(04):883–914, 2004.

Marcus, Gary F. Negative evidence in language acquisition. *Cognition*, 46(1):53–85, 1993.

Marsh, Allison. Tracking the puma. In *Proceedings of the IEEE Conference on the History of Electronics*, 2004.

Maturana, Humberto Romecin. *Autopoiesis and cognition: The realization of the living*, volume 42. Springer, 1980.

Meeden, Lisa A and Blank, Douglas S. *Introduction to developmental robotics*. Taylor & Francis, 2006.

Meltzoff, Andrew N. Infant imitation after a 1-week delay: Long-term memory for novel acts and multiple stimuli. *Developmental Psychology*, 24(4):470, 1988.

Meltzoff, Andrew N. The 'like me' framework for recognizing and becoming an intentional agent. *Acta psychologica*, 124(1):26–43, 2007a.

Meltzoff, Andrew N. 'like me': a foundation for social cognition. *Developmental science*, 10(1):126–134, 2007b.

Meltzoff, Andrew N and Keith Moore, M. Imitation, memory, and the representation of persons. *Infant behavior and development*, 17(1):83–99, 1994.

Meltzoff, Andrew N and Moore, M Keith. Imitation of facial and manual gestures by human neonates. *Science*, 198(4312):75–78, 1977.

Meltzoff, Andrew N and Prinz, Wolfgang. *The imitative mind: Development, evolution and brain bases*, volume 6. Cambridge University Press, 2002.

Metta, Giorgio, Sandini, Giulio, Vernon, David, Natale, Lorenzo, and Nori, Francesco. 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.

Morales, Michael, Mundy, Peter, Delgado, Christine EF, Yale, Marygrace, Messinger, Daniel, Neal, Rebecca, and Schwartz, Heidi K. Responding to joint attention across the 6-through 24-month age period and early language acquisition. *Journal of Applied Developmental Psychology*, 21(3):283–298, 2000.

Moran, Michael E. Evolution of robotic arms. *Journal of Robotic Surgery*, 1(2):103–111, 2007.

Morgan, James L and Demuth, Katherine. Signal to syntax: An overview. *Signal to syntax: Bootstrapping from speech to grammar in early acquisition*, pages 1–22, 1996.

Mundy, Peter, Block, Jessica, Delgado, Christine, Pomares, Yuly, Van Hecke, Amy Vaughan, and Parlade, Meaghan Venezia. Individual differences and the development of joint attention in infancy. *Child development*, 78(3):938–954, 2007.

Musa, Banu. *The Book of Ingenious Devices*. Springer, translated by Donald Routledge Hill, 1979.

Nadajaran, Gunalan. Islamic Automation: A Reading of al-Jazari'sThe Book of Knowledge of Ingenious Mechanical Devices. In *The First International Conference on the Histories of Art, Science and Technology*, 2008.

Nagell, Katherine, Olguin, Raquel S, and Tomasello, Michael. Processes of social learning in the tool use of chimpanzees (pan troglodytes) and human children (homo sapiens). *Journal of Comparative Psychology*, 107(2):174, 1993.

Naigles, Letitia. Children use syntax to learn verb meanings. *Journal of child language*, 17(02):357–374, 1990.

Neisser, Ulric. Five kinds of self-knowledge. *Philosophical psychology*, 1(1):35–59, 1988.

Nelson, Katherine and Fivush, Robyn. The emergence of autobiographical memory: a social cultural developmental theory. *Psychological review*, 111(2):486, 2004.

Nielsen, Mark. Copying actions and copying outcomes: social learning through the second year. *Developmental psychology*, 42(3):555, 2006.

Nielsen, Mark, Simcock, Gabrielle, and Jenkins, Linda. The effect of social engagement on 24-month-olds imitation from live and televised models. *Developmental science*, 11 (5):722–731, 2008.

Perregaux, Charles and Perrot, F. Louis. *Les Jaquet-Droz et Leschot*. Attinger Frères, 1916.

Pfeifer, Rolf et al. *Understanding intelligence*. The MIT Press, 2001.

Piaget, Jean. *Le jugement moral chez l'enfant*. F. Alcan Paris, 1932.

Pinker, Steven. *The language instinct: How the mind creates language*. HarperCollins, 2010.

Piolino, P, Hisland, M, Ruffeveille, I, Matuszewski, V, Jambaqué, I, and Eustache, F. Do school-age children remember or know the personal past? *Consciousness and Cognition*, 16(1):84–101, 2007.

Poupyrev, Ivan, Nashida, Tatsushi, and Okabe, Makoto. Actuation and tangible user interfaces: the vaucanson duck, robots, and shape displays. In *Proceedings of the 1st international conference on Tangible and embedded interaction*, pages 205–212. ACM, 2007.

Povinelli, Daniel J, Landau, Keli R, and Perilloux, Helen K. Self-recognition in young children using delayed versus live feedback: Evidence of a developmental asynchrony. *Child development*, 67(4):1540–1554, 1996.

Prepin, Ken and Gaussier, Philippe. How an agent can detect and use synchrony parameter of its own interaction with a human? In *Development of Multimodal Interfaces: Active Listening and Synchrony*, pages 50–65. Springer, 2010.

Prince, Christopher G and Demiris, Yiannis. Editorial: introduction to the special issue on epigenetic robotics. *Adaptive Behavior*, 11(2):75–77, 2003.

Quine, WV. Word and object, 1960.

Riskin, Jessica. The defecating duck, or, the ambiguous origins of artificial life. *Critical Inquiry*, 29(4):599–633, 2003.

Rosch, Eleanor H. On the internal structure of perceptual and semantic categories. 1973.

Rosheim, Mark Elling. *Robot Evolution: The Development of Anthrobotics*. 1st edition. London, 1994.

Rosheim, Mark Elling. *Leonardo's Lost Robots*. New York: Springer, foreword by Carlo Pedretti, 2006.

Ruby, Emily, Black, Samuel, and Ciotola, Nicholas. Adventures in innovation 1920-1945. *Western Pennsylvania History*, 92(1):40–51, 2009.

Salichs, M and Balaguer, Carlos. The current state of robotics: Post or pre-robotics? In *Proc. IARP 3rd International Workshop on Service, Assistive and Personal Robots*, pages 79–84. IARP, 2003.

Sampson, Geoffrey. Exploring the richness of the stimulus. *The Linguistic Review*, 18 (1-2):73–104, 2002.

Scassellati, Brian Michael. *Foundations for a Theory of Mind for a Humanoid Robot*. PhD thesis, Massachusetts Institute of Technology, 2001.

Seth, Anil Kumar. *On the relations between behaviour, mechanism, and environment: Explorations in artificial evolution*. University of Sussex, 2000.

Sharkey, N and Sharkey, A. Electro-mechanical robots before the computer. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 223(1):235–241, 2009.

Singh, Balkeshwar, Sellappan, N, and Kumaradhas, P. Evolution of industrial robots and their applications. *International Journal of Emerging Technology and Advanced Engineering*, 3(5):763–768, 2013.

Spillemaecker, Chanta and al. *Vaucanson et l'homme artificiel: des automates aux robots*. Presses universitaires de Grenoble, 2010.

Steels, Luc. Language games for autonomous robots. *Intelligent Systems, IEEE*, 16(5): 16–22, 2001.

Steels, Luc and Baillie, Jean-Christophe. Shared grounding of event descriptions by autonomous robots. *Robotics and autonomous systems*, 43(2):163–173, 2003.

Stich, Stephen P. Empiricism, innateness, and linguistic universals. *Philosophical Studies*, 33(3):273–286, 1978.

Stoytchev, Alexander. Some basic principles of developmental robotics. *Autonomous Mental Development, IEEE Transactions on*, 1(2):122–130, 2009.

Struijk, Bob. Robots in human societies and industry. *AARMS*, 10(1), 2011.

Thiessen, Erik D and Saffran, Jenny R. When cues collide: use of stress and statistical cues to word boundaries by 7-to 9-month-old infants. *Developmental psychology*, 39(4): 706, 2003.

Thompson, Evan. Sensorimotor subjectivity and the enactive approach to experience. *Phenomenology and the Cognitive Sciences*, 4(4):407–427, 2005.

Tomasello, Michael. Cultural transmission in the tool use and communicatory signaling of chimpanzees? 1990.

Tomasello, Michael. The item-based nature of children's early syntactic development. *Trends in cognitive sciences*, 4(4):156–163, 2000.

Tomasello, Michael. *Origins of human communication*. MIT press Cambridge, 2008.

Tomasello, Michael. *The cultural origins of human cognition*. Harvard University Press, 2009.

Tomasello, Michael and Farrar, Michael Jeffrey. Joint attention and early language. *Child development*, pages 1454–1463, 1986.

Tomasello, Michael, Kruger, Ann C, and Ratner, Hilary Horn. Cultural learning. *Behavioral and brain sciences*, 16:495–495, 1993.

Tomasello, Michael, Carpenter, Malinda, Call, Josep, Behne, Tanya, Moll, Henrike, et al. Understanding and sharing intentions: The origins of cultural cognition. *Behavioral and brain sciences*, 28(5):675–690, 2005.

Trevarthen, Colwyn. Communication and cooperation in early infancy: A description of primary intersubjectivity. *Before speech: The beginning of interpersonal communication*, pages 321–347, 1979.

Tulving, Endel. Episodic and semantic memory1. *Organization of memory*, pages 381–402, 1972.

Tulving, Endel. Remembering and knowing the past. *American Scientist*, 77(4):361–367, 1989.

Tulving, Endel. Episodic memory: From mind to brain. *Annual review of psychology*, 53 (1):1–25, 2002.

Tulving, Endel, Schacter, Daniel L, McLachlan, Donald R, and Moscovitch, Morris. Priming of semantic autobiographical knowledge: a case study of retrograde amnesia. *Brain and cognition*, 8(1):3–20, 1988.

Turing, Alan M. Computing machinery and intelligence. *Mind*, 59(236):433–460, 1950.

Valavanis, K. P., Vachtsevanos, G. J., and Antsaklis, P. J. Technology and Autonomous Mechanisms in the Mediterranean: From Ancient Greece to Byzantium. In *Proceedings of the European Control Conference*, Kos, Greece, 2007.

Van Schaik, Carel P, Ancrenaz, Marc, Borgen, Gwendolyn, Galdikas, Birute, Knott, Cheryl D, Singleton, Ian, Suzuki, Akira, Utami, Sri Suci, and Merrill, Michelle. Orangutan cultures and the evolution of material culture. *Science*, 299(5603):102–105, 2003.

Varela, Franscisco J, Thompson, Evan T, and Rosch, Eleanor. *The embodied mind: Cognitive science and human experience*. The MIT Press, 1991.

Vernon, David. Enaction as a conceptual framework for developmental cognitive robotics. *Paladyn*, 1(2):89–98, 2010.

Vernon, David, Metta, Giorgio, and Sandini, Giulio. A survey of artificial cognitive systems: Implications for the autonomous development of mental capabilities in computational agents. *Evolutionary Computation, IEEE Transactions on*, 11(2):151–180, 2007.

von Hofsten, Claes and Fazel-Zandy, Shirin. Development of visually guided hand orientation in reaching. *Journal of experimental child psychology*, 38(2):208–219, 1984.

Voskuhl, Adelheid. *Androids in the Enlightenment: Mechanics, Artisans, and Cultures of the Self*. University of Chicago Press, 2013.

Vygotskij, Lev S. *Thought and language*. MIT press (editon of 2012), 1934.

Walter, W Grey. A machine that learns. *Scientific American*, 185(2):60–63, 1951.

Walter, William Grey. An imitation of life. *Scientific American*, 182(5):42–45, 1950.

Warneken, Felix and Tomasello, Michael. Helping and cooperation at 14 months of age. *Infancy*, 11(3):271–294, 2007.

Warneken, Felix, Chen, Frances, and Tomasello, Michael. Cooperative activities in young children and chimpanzees. *Child development*, 77(3):640–663, 2006.

Weng, Juyang, McClelland, James, Pentland, Alex, Sporns, Olaf, Stockman, Ida, Sur, Mriganka, and Thelen, Esther. Autonomous mental development by robots and animals. *Science*, 291(5504):599–600, 2001.

Whiten, Andrew. Primate culture and social learning. *Cognitive Science*, 24(3):477–508, 2000.

Whiten, Andrew and Van Schaik, Carel P. The evolution of animal 'cultures' and social intelligence. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 362 (1480):603–620, 2007.

Whiten, Andrew, Goodall, Jane, McGrew, William C, Nishida, Toshisada, Reynolds, Vernon, Sugiyama, Yukimaru, Tutin, Caroline EG, Wrangham, Richard W, and Boesch, Christophe. Cultures in chimpanzees. *Nature*, 399(6737):682–685, 1999.

Whiten, Andrew, McGuigan, Nicola, Marshall-Pescini, Sarah, and Hopper, Lydia M. Emulation, imitation, over-imitation and the scope of culture for child and chimpanzee. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 364(1528):2417–2428, 2009.

Wilson, Margaret. Six views of embodied cognition. *Psychonomic bulletin & review*, 9(4): 625–636, 2002.

Wood, David, Bruner, Jerome S, and Ross, Gail. The role of tutoring in problem solving*. *Journal of child psychology and psychiatry*, 17(2):89–100, 1976.

Xagoraris, Zafirios. *The Automaton Theater*. Massachusetts Institute of Technology, 1991.

Zlatev, Jordan and Balkenius, Christian. Introduction: Why epigenetic robitcs? In *Proc. 1st Int. Workshop Epigenetic Robot.*, 2001.