Cognitive Architecture for Mutual Modelling

Alexis Jacq[1,2], Wafa Johal[1], Pierre Dillenbourg[1], Ana Paiva[2]
[1]CHILI Lab, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland
[2]INESC-ID & Instituto Superior Técnico, University of Lisbon, Portugal

***Abstract*—In an educational scenario, learners and teacher are able to understand each other. Learners display behaviours in order to show their understanding and teachers adapt in order to make sure that the learners' knowledge is the required one. This ability requires a model of their own mental states perceived by others.**

In this paper, we discuss of the importance of a cognitive architecture enabling second-order Mutual Modelling for Human-Robot Interaction in educative contexts.

I. Introduction

A social robot is brought to interact with humans. The quality of this interaction depends on its ability to behave in an acceptable and understandable manner by the user. Hence the importance for a robot to take care of his image: how much it is perceived as an automatic and boring agent, or contrariwise as a surprising and intelligent character. If the robot is able to detect this perception of itself, it can adapt its behaviour in order to be understood: "you think I am sad while I am happy, I want you to understand that I am happy". In a collaborative context, where knowledge must be shared, agents must exhibit that they are acquiring the shared information with an immediate behaviour: "I look what you are showing to me, do you see I am looking what you are showing, do you think I am paying attention to your explanation ?"; "I have understood your idea, do you understand I have understood ?". We have different strategies to exhibit understanding or to repair a miss-understanding. As an example, if someone is talking about a visual object, we alternatively gaze at the object and at the person to make sure he saw we gazed at the object. Or if we detect that the other have not understood a gesture (e.g. pointing an object) we probably exaggerate the gesture.

Developed by Baron-Cohen and Leslie [**?**], the Theory of Mind (ToM) describes the ability to attribute mental states and knowledge to others. In interaction, humans are permanently collecting and analysing huge quantity of information to stay aware of emotions, goals and understandings of their fellows. In this work, we focus on a generalization of this notion: Mutual Modelling characterizes the effort of one agent to model the mental state of another one [**?**].

Until now, the work conduced by the Human-Robot Interaction (HRI) community to develop mutual modelling abilities in robots was limited to a first level of modelling (see Related Work section **??**). Higher levels require the ability to recursively attribute a theory of mind to other agent (*I think that you think that ...*) and their application to HRI remains unexplored. However, a knowledge of oneself perceived by others is necessary to adapt a behaviour to keep mutual understandings. Without such a reflection it is possible to try collaborations, but no immediate feedbacks of success are given. If the trial has failed it is impossible to guess it and to repair it.

An important challenge of social robotics is to provide assistance in education. The ability of robots to support adaptive and repetitive task can be valuable in a learning interaction. The CoWriter Project [**?**], [**?**] introduces a new approach to help children with difficulties to learn handwriting. Based on the *learning by teaching* paradigm, the goal of the project is not only to help children with their handwriting but mainly to improve their self-confidence and motivation in practising handwriting.

Learning by teaching engages students to conduct the activity in the role of the teachers in order to support their learning process. This paradigm is known to produce motivational, meta-cognitive and educational benefits in a range of disciplines [**?**]. The CoWriter project is the first application of learning by teaching approach to handwriting.

The effectiveness of this learning by teaching activity is built on the "protégé effect": the teacher feels responsible for his student, commits to the student's success and possibly experiences student's failure as his own failure to teach. The main idea is to promote the child's extrinsic motivation to write letters (he will do it in order to help his "protégé" robot) and to reinforce the self-esteem of the child (he plays the teacher and the robot actually progresses).

In that context, the robot needs to pretend enough difficulties to motivate the child to help it. This pretending strongly depends on the perception of the robot by the child: the displayed behaviours (gestures, gazes and sounds) by the robot, the initial level and learning speed of the robot must match with what the child imagines of a "robot in difficulty". In order to adapt to the child, the robot needs then to have a model of how it is perceived by the child. On the other side, the child builds also a model of the robot's difficulties and attitude. This mutual-modelling is primordial in order to have mutual understanding and fluid interaction between learner and teacher.

II. Related works

A large amount of fields have introduced frameworks to describe mutual modelling ability [**?**]. In developmental psychology Flavell [**?**] denotes two different levels of perspective taking: the *cognitive connection* (I see, I hear, I want, I

like...) and *mental representation* (what other agents feel, hear, want...).

From a computational perspective, Epistemic logic describes knowledges and beliefs shared by agents. This framework enables consideration of infinite-level of mutual modelling. It defines a *shared-knowledge* (all the agents of a group know **X**) and a *common-knowledge* (all the agents of a group know **X**, and know that all the agent know **X**, and know that all the agents know that all the agents know **X**, ...) [**?**].

Mutual modelling has also been studied through educational contexts. Roschelle and Teasley [**?**] suggested that collaborative learning requires a *shared understanding* of the task and of the shared information to solve it. The term "mutual modelling" was introduced in Computer-Supported Collaborative Learning (CSCL) by Dillenbourg [**?**]. It focused on knowledge states of agents. Dillenbourg developed in [**?**] a computational framework to represent mutual modelling situations.

However, HRI research has not, until now, explored the whole potential of mutual modelling. In [**?**], Scassellati supported the importance of Leslie's and Baron-Cohen's theory of mind to be implemented as an ability for robots. He focused his work on attention and perceptual processes (face detection or colour saliency detection). Thereafter, some works (including Breazeal [**?**], Trafton [**?**], Ros [**?**] and Lemaignan [**?**]) were conduced to implement Flavell's first level of perspective taking [**?**] ("*I see (you do not see the book)*"), ability that is still limited to visual perception.

Breazeal [**?**] and Warnier [**?**] reproduced the Sally and Anne's test of Wimmer [**?**] with robots able to perform visual perspective taking. The robot was able to infer the knowledge of a human given the historic of his visual experience.

In [**?**], Lemaignan implemented a system that compute in real-time the visual field of agents and estimate which objects are looked at. This time, the robot is not just aware of what *can be seen* by agents, but it perceives what *is currentl y looked at*. Lemaignan used this system to measure Sharma's *with-me-ness* [**?**], visual commitment based on expected focus of attention in an activity.

III. MM-BASED REASONING

We must make a distinction between architectures that performs mutual modelling and simple cognitive architectures (that does not model other agents) in order to avoid miss-understandings and Russel's paradoxes. Let's call *basic* such a simple architecture and *MM-based* the mutual modelling ones.

A first intuition for mutual modelling is to assume that all agents have the same basic architecture. In [], Breazeal show a MM-based reasoning where the robot uses its own basic architecture to model other agents. We can imagine a second level of modelling where a robot uses its MM-based architecture to model the other agent. But that creates an infinite recursive loop: the agent then models the robot that models the agent etc. Another reason to avoid a recursive approach is that different agent must have different behaviours: in similar situations, they will not necessary take similar decisions.

We propose a different approach of modelling, where we define two orders of agents: the *first-order-agents* concern direct representations of agents by the robot (for example the child), while the *second-order-agents* concern the representation of agents by agents (for example the robot-perceived-by-the-child). To model second-order-agent like the robot-perceived-by-the-child will help to model how the child perceives the robot, e.g. to make sure the child understand that the robot is learning from his demonstrations. We can as well define n^{th}*-order* agents with higher level of theory of mind. But taking into account high levels of mutual modelling would be difficult to process in real time. Unlike the epistemic logic, our proposed framework will not take into account infinite regress [**?**] of mutual modelling.

All sensors (cameras, micros, motor positions ... and in the case of CoWriter the tablet's inputs) are used to perceive information about the physical behaviour of agents. We call *perceived variables* all the measurable quantities or qualities that provide such information (position in space, gaze's direction, speeches, movements, facial expressions ...). Thus, each agent's model is associated with a set of perceived variables that describes his physical behaviour.

But emotional states of agents cannot be directly measured directly from sensors. We call *abstract variables* all the quantities or qualities that describe the mental state of an agent. Abstract variables are deduced from the dynamic of perceived variables. As an example, if the robot shows an object by pointing it with its arm, it expects the child to look at the object. If then the child looks at the hand of the robot, the robot can deduce that the child has not understood the meaning of its gesture. The perceived variables are the robot's gesture and the gaze direction of the child. The deduced abstract variable is the understanding of the gesture by the child.

A model of an agent is the set of all the values of the perceived or abstract variables associated with this model. Since the values of variables are likely to change with the time, the models must be dynamic.

In order to deduce the values of abstract variables (that can't be obtained from direct perception), we propose to build a Bayesian model based on the knowledge from perceived variables. The choice of a probabilist approach instead of a symbolic approach comes from the errors in the perception of the robot: knowledge and other mental states of agents can not be directly perceived through the behaviours. They must be inferred, hence a probabilistic model enables richer predictions.

This Bayesian network would contain the probabilities that *abstract variables* take values given the values of *perceived variables*.

For example, if the robot points at an object and detects that a child saw the movement, then it expects that the child will look at the targeted object. In other terms, if the child looks at the hand of the robot but does not look at the target object in the following instants, the probability that the child understood the pointing movement is expected to be small.

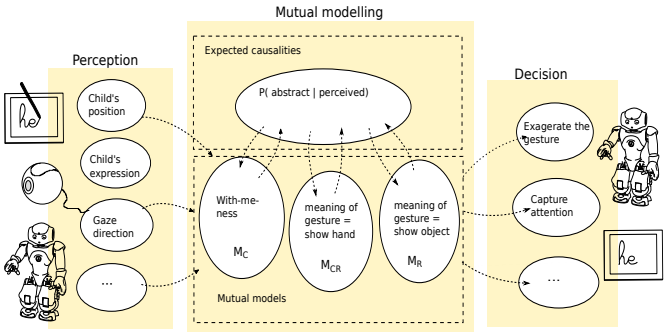

Fig. 1. **Overview of the cognitive architecture**. Yellow squares represent different main parts. White ellipses represent modules. We named in example some content of modules and illustrated possible devices used for perception and decisions.

Knowing that, the robot can make the decision to exaggerate its movement of pointing.

IV. Description of the architecture

The picture 1 visually summarize the global design of our architecture.

Our cognitive architecture for mutual modelling can be designed in three main parts. The **perception part** (see IV-A) will regroup all the modules that measure the values of the perceived variables using sensors. These values are sent to the **mutual modelling part** (see IV-B) that updates in real-time models with measured values of perceived variables and deducts the value of abstract variables. Finally, the **decision part** (see IV-C) contains all the modules associated to the control of the robot (and other active devices like tablets in CoWriter). These modules can read values given by mutual models in order to compute decisions. In the example of the CoWriter activity, these modules are given by the system that learns and generate letters, but we can add a module that generates micro-behaviour, another that decides to switch to a new activity (e.g. drawing with the robot),.... The following subsections explain in detail the content and operation of each part of the architecture.

A. Perception modules

The sensitive modules contain all the module able to use sensitive devices to measure value of relevant perceived variables. Some of these modules are associated with the content of the interaction activity. In CoWriter, the learning module takes inputs of tablets to compute the new state of robot's writing. It defines a sensitive module, and the value of the new state of the robot to write a letter defines a perceived variable. As well, the evaluation of the robot by the child via the feedback buttons defines another perceived variable provided by the modules of the activity. Other modules are independent of the activity: the system that estimates the target objects looked by the child provides additive information not directly used by the modules of the activity.

B. Mutual modelling modules

Each perceived value measured by sensitive modules are associated with the model of an agent (or a n^{th}-order agent). Each mutual model can be designed as a module that knows the list of its associated perceived variables and watch if the value of one of these variables has been changed. An additive module knows all the expected causalities and computes the values of abstract variables. Some expected causalities can be empirically learned and other set by hands.

C. Decision making

We believe that the values of variables provided by mutual modelling will provide rich and useful information for decision making. Taking in account these values to elaborate decision should improve the realism and the efficiency of the robot in the interaction. Just like the sensitive ones, the modules that take decision can be directly associated to the activity (in CoWriter, the choice of a new learning curve or the decision to suddenly make a big mistake), or can govern independent behaviour (for example a module that generate the micro-behaviour of the robot).

Some decisions, especially the one that will govern the micro-behaviours of the robot can be autonomously made by the robot. It will not strongly affect the content and objectives of the activity (in CoWriter the main objective is to provide the child with a new extrinsic motivation to write in helping the robot). But other decisions can have a high impact on the progress of the interaction: to stop an activity and to switch to a new one can frustrate a child that was committed into the activity. The conditions to make such a decision are not directly assessable, but must be learned by the robot. In order to make these decision cautiously, we propose to start by a Wizard-of-Oz approach and to move towards an autonomous approach following these steps:

1) **Wizard-of-Oz**: A human takes decisions; the robot learns
2) **Mixed-initiative**: The robot makes suggestions; a human agrees or disagrees
3) **Autonomous**: The robot makes decisions

V. Conclusion

Educational HRI based on learning by teaching approach needs robot able to perform second-level mutual modelling. We introduced a new approach to implement mutual modelling into a cognitive architecture. We used the CoWriter activity as an example of application, but our architecture could be easily generalised for any kind of interaction. We believe that this step must be reached in other context of HRI, in order to develop higher realism of behaviours and to improve the quality of interactions.