A Novel Approach for Behavior Management and Real-Time Adaptation during Child-Robot Interaction

Giulia Berettieri[1,†], Anna Allegra Bixio[1,*,†], Lucrezia Grassi[1], Carmine Recchiuto[1] and Antonio Sgorbissa[1]

[1]*Dibris Dept., University of Genoa, Genoa, Italy*

Abstract

The paper presents a robotic architecture for managing behavior in educational settings, inspired by Applied Behavior Analysis (ABA). The system features a social robot that (1) learns a model of a child's goals, beliefs, and intentions, grounded in observations and conversations, to explain their detected behavior, and (2) responds effectively to behaviors by planning appropriate sequences of actions to implement the strategies suggested by experts. By leveraging cloud-based processing and local execution, the robot dynamically adapts to social interactions in real-time, delivering educational activities, monitoring children's behavior, and applying behavior management strategies. Results from the experimental evaluations highlight the system's replanning and cloud response times, as well as its overall effectiveness.

Keywords

Behavior Management, Real-Time Adaptation, Child-Robot Interaction

1. Introduction

We introduce a novel robotic architecture for behavior management in educational environments, drawing inspiration from Applied Behavior Analysis (ABA), an evidence-based framework that explains how behavior is influenced by environmental factors [1]. While the use of ABA in supporting individuals with neurodevelopmental disorders remains a subject of debate [2], we propose that an ABA-inspired robot, capable of understanding the purpose of others' behaviors could enhance the quality of child-robot interactions (CRI) in learning environments.

Robots in education currently employ computational methods to tailor learning activities to students' individual needs [3]. Recent studies on behavior change strategies using social robots have shown that motivational approaches and design features of robots positively influence children's adoption of healthier behaviors [4]. Specifically, in child-robot interactions based on ABA principles, SARs act as mediators to enhance social skills in individuals with Autism Spectrum Disorder (ASD) [5].

Investigations in this field explore how Theory of Mind (ToM)—the ability to understand others' mental states [6]—can be computationally modeled [7] and how impacts human trust and decision-making [8].

The contributions of this work are the following:

- We developed a theoretical model that takes into account the constraints in CRI, serving as the robotic counterpart to the ABA model.
- We developed an architecture based on the robotic ABA model we previously created. It enables real-time adaptation and allows the robot to (1) infer child's mental state, grounded in observations and conversations, and (2) plan appropriate actions using the Planning Domain Definition Language (PDDL).
- We evaluated the system's performance in terms of planning and response time.

Workshop on Advanced AI Methods and Interfaces for Human-Centered Assistive and Rehabilitation Robotics (a Fit4MedRob event) - AIxIA 2024

*Corresponding author.

†These authors contributed equally.

s4816056@studenti.unige.it (G. Berettieri); annaallegra.bixio@edu.unige.it (A. A. Bixio); lucrezia.grassi@edu.unige.it (L. Grassi); carmine.recchiuto@dibris.unige.it (C. Recchiuto); antonio.sgorbissa@unige.it (A. Sgorbissa)

Section 2 introduces the ABA methods and describes the scenario. Section 3 details the system's architecture and behavioral planning. Section 4 discusses a case study of behavior function recognition. Section 5 summarizes the results. Finally, Section 6 discusses the conclusions and limitations.

2. Background and problem statement

Applied Behavior Analysis is an evidence-based approach aimed at improving socially significant behaviors [1]. In the ABA framework, behavior is modeled as purposeful (functional) and influenced by stimuli occurring before (antecedents) and after the behavior (consequences). Functional Behavior Assessment (FBA) [9] is the process of identifying the functions or purposes of behavior, through an assessment of its antecedent and consequence. The study considers two functions:

- *Gain a Tangible*, i.e. the purpose of behaviors aiming at having access to preferred stimuli.
- *Escape*, i.e. the purpose of behaviors aiming at avoiding unpleasant situations.

FBA also provides valuable support to educators by offering a structured method for managing complex behaviors in a classroom setting.

Given this premise, the problem we aim to address is how to develop an autonomous social robot capable of planning educational activities, continuously monitoring children's actions, conducting FBA to identify the purposes of challenging behaviors, and proposing alternative strategies to handle them. Our goal is not to categorize behaviors but to address a common classroom struggle: managing behaviors considered “challenging” because they present major challenges to maintaining a productive learning environment. A robot that can tailor learning interactions and respond effectively to others' behaviors through the use of effective strategies could greatly improve the integration of robotics in education.

3. System architecture and planning for behavioral change

To achieve effective and autonomous behavior management in CRI, we have designed a software architecture that integrates cloud-based processing with local execution capabilities. Figure 1 illustrates the detailed layout of the proposed system architecture, which is divided into two main components: the Server and the Client.

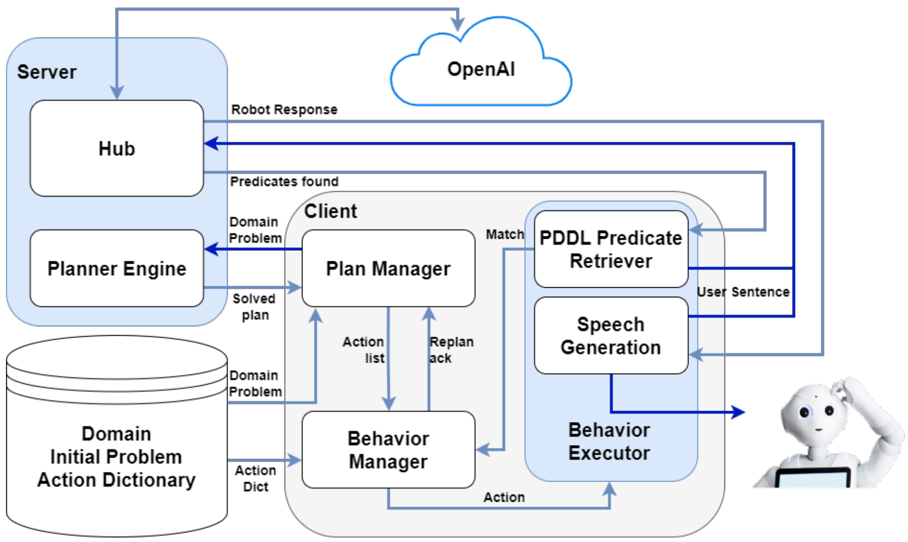


Figure 1: System architecture integrating cloud services with local execution.

The server hosts two primary elements: the **Hub** and the **Planner Engine**. The **Hub** manages connections to the external cloud service OpenAI, while the Planner Engine leverages Fast Downward, a highly efficient planning system based on the Problem Domain Definition Language. The **Planner Engine** processes inputs from the client, specifically the PDDL domain and the initial problem. These files outline the available actions, their effects, preconditions, and the initial state of the problem that the robot needs to address.

The client, embedded within the robot, comes preloaded with the necessary domain and initial problem files. The **Plan Manager** is responsible for sending these files to the **Planner Engine** and retrieving the generated plan.

Upon receiving the plan, the **Behaviour Manager** takes charge, iterating through the plan and determining the actions that need to be executed by checking the predicates.

As actions are performed, the **PDDL Predicate Retriever** asserts new predicates about the current state of the world based on the interaction with the child and the environment. The goal is to monitor plan progress to ensure that the actual preconditions and effects of actions match the expected ones.

On one hand, the **PDDL Predicate Retriever** employs its sensors to make assertions about the current state of the world; on the other hand, it attempts to infer the child's mental state through conversation. To this end, the robot employs Microsoft Azure services to transcribe the child's spoken words captured by microphones. The transcribed text is then sent to OpenAI, which analyzes the content to generate appropriate responses and identify predicates related to the child's behavior.

The **Behavior Manager** compares asserted predicates with the expected preconditions and effects in the plan. If the predicates do not match, the **Behavior Manager** triggers the **Plan Manager**, which may then initiate a replan. This process follows a *match and go* approach, where the robot continuously evaluates the environment against a set of predefined predicates and conditions that must be met for an action to be executed. When a new situation arises, the robot checks its current state against the effects of the current action and all other actions' preconditions. If these conditions *match* the expected criteria, the robot can either proceed to execute the next action — hence the term *go* — or request a replan. This method ensures that the robot's actions are contextually adaptive.

4. A case study: *Escape* and *Gain a Tangible*

Drawing from ABA and FBA principles, two behavior functions were identified to be recognized for preliminary testing of the model's validity: *Gain a Tangible* and *Escape*. The aim is that, through verbal interaction with the child, the robot would be able to autonomously recognize the occurrence of the following behaviors:

- *Gain a Tangible*: When pleasant activities are interrupted by time limits, negative reactions of the child may be elicited.
- *Escape*: When presented with a task, a child may perceive it as too difficult or too boring, potentially provoking adverse responses to avoid the task.

With a comprehensive understanding of these psychological approaches, we developed a PDDL domain capable of autonomously managing various scenarios of interest according to the ABA approach. Leveraging actions detailed in Table 1, the framework can adapt to different situations, including making real-time adjustments. The initial plan involves a robot presenting a memory card game to a child. The process begins with a brief interaction (`Interact`), followed by an explanation of the rules (`PresentTask`).The robot then monitors the child while they play (`Wait`). As time nears its end, the robot warns the child (`WarningCleanUp`), instructs them to stop and put away the cards (`PutAway`), confirms they have finished (`ConfirmAllGood`), and ends with praise (`GoodJob`). During the `Wait` action, the robot may detect issues by querying students via OpenAI-supported verbal interactions.

The paragraphs below present some examples of verbal interaction logs related to this situation. User inputs are denoted by "U," the robot's responses by "R," and the retrieved predicates by "P."

U: Pepper, can we talk while I am playing?
R: Giulia, let's play together!

Action	Description
`DiscoverTOM`	Prompt dialogue to assess child's ToM
`GoodJob`	Praise child for achievement
`Interact`	Engaging conversation with the child
`PresentTask`	Explain task rules in detail
`PutAway`	Ensure the child stops playing
`StrategyHardTask`	Apply strategy if child finds task hard
`StrategyWant2Play`	Apply strategy if child still wants to play
`Wait`	Wait and monitor task execution
`WarningCleanUp`	Warn about time and monitor completion

Table 1
PDDL Actions related to Behavior Management.

P: `warning, not_waiting, want_to_play`
U: No, don't worry, I don't want to play anymore.
R: Giulia, why are you so upset?
P: `warning, not_waiting, want_to_play, not_agreed, not_doing`
Replan is needed

In the exchange above, the robot detects a shift in the child's engagement, recognizing that Giulia is no longer interested in continuing the game. This prompts replanning starting from the current state (see Section 3), with the asserted predicate `not_doing` acting as a trigger. In response to this interaction, a new plan is formulated, which includes the action `(DiscoverToM1 game kid memory)`. The effect of this action is `investigating`, and the following log shows a possible outcome of the interaction.

U: Because this game is too difficult!
R: Giulia, what makes this game difficult for you?
P: `investigating, task_is_hard`
Replan is needed

This update leads to the conclusion that the task is perceived as too challenging for the user, triggering again the **PDDL Predicate Retriever** to update the predicates describing the current state. It causes a mismatch with the expected effects of the current action that is detected by the **Behavior Manager**, which prompts the **Plan Manager** for a replanning process that will generate a new sequence of actions. This particular case would confirm that the child believes the proposed task is too difficult, and the robot will respond accordingly by implementing the *Escape* function strategy. However, this is just one possible scenario.

5. Performance Evaluation

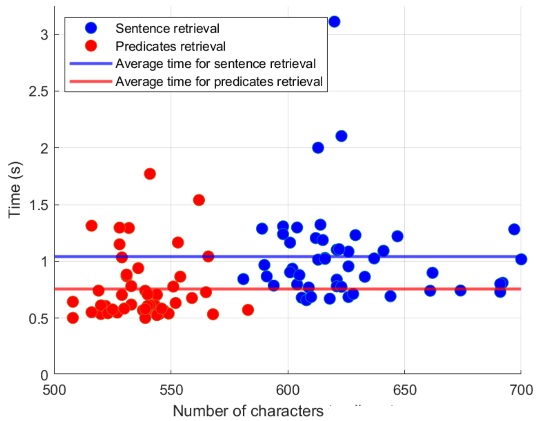
To assess our model's effectiveness in educational settings, we have developed an experimental design comprising one-on-one child-robot interaction sessions, where the children will engage in a memory game task. The robot's accuracy in detecting behavior functions and the appropriateness of employed strategies will be retrospectively evaluated by educational experts.

The experimental evaluation of the framework described in Sections 3 and 4, is conducted in a controlled laboratory setting focused on a memory game task (Fig.2b). The setup involves a one-on-one interaction between one of the authors and the humanoid robot Pepper. The experiment is repeated multiple times with the same participant to test various scenarios of replanning within the framework. The objective is to observe the triggering conditions for each scenario and evaluate whether the framework can accurately assert the relevant grounded predicates at the appropriate moments. To this end, key metrics being assessed include:

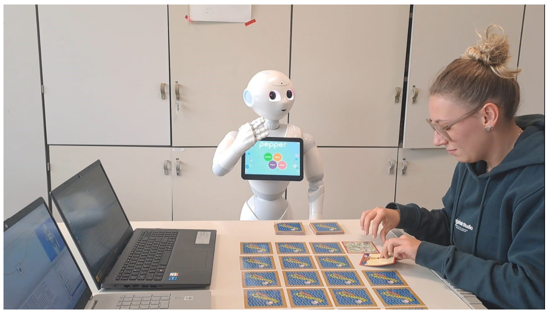
- *Replan Time*: Measures the time from the introduction of a disruption to the completion of the replanning process.
- *Cloud Response Time*: Evaluates the time taken for the robot to send a request, process cloud information, and receive a response.

Concerning *Replan time*, cloud resources were compared with local planning by sending the same PDDL problem to both a cloud-based Planner Engine and a local one. Planning times for the cloud range from 0.09 to 0.13 seconds, while local times range from 0.08 to 0.12 seconds. This indicates that cloud resources do not reduce planning time, likely due to the simplicity of the generated plans.

Regarding the *Cloud Response Time*, the graph in Figure 2a shows both the time required to retrieve a response sentence used to reply to the user (blue) and the time taken to retrieve grounded predicates (red).



(a) Cloud Response Time vs. Number of Characters in the User's Sentence.



(b) Detecting *Escape* and *Gain a Tangible* during a robot-assisted memory card game.

Figure 2: Graph and Picture of Performance Evaluation.

The Figure 2a illustrates a difference in character processing between the two tasks. Our implementation involves more complex instructions sent to OpenAI, incorporating additional context and user-specific parameters. This leads to varying prompt lengths, impacting the average cloud response time: approximately 1 second for sentence retrieval and 0.75 seconds for predicate retrieval. Both times are significantly below the 2-second threshold deemed acceptable for human conversation [10] [11], although occasional delays of ≈ 3 seconds occur.

6. Conclusion

Our study introduces a framework for enhancing child-robot interactions using behavior analysis principles. The system architecture includes an innovative framework for online replanning to adapt dynamically to user behavior. However, it is important to acknowledge several limitations. In its current initial version, our model does not account for children's acceptance of the robot, their motivation, or attention during interactions. Addressing these factors is crucial for conducting more precise functional analyses. Future research should focus on optimizing our model by incorporating these elements to improve its effectiveness and acceptance.

Acknowledgment

This work has been co-funded by Ministero dell'Università e della Ricerca, Piano Nazionale Complementare (PNC), FIT4MEDROB project, Mission 3.

References

[1] J. O. Cooper, T. E. Heron, W. L. Heward, et al., Applied behavior analysis, Pearson/Merrill-Prentice Hall Upper Saddle River, NJ, 2007.

[2] J. B. Leaf, J. H. Cihon, R. Leaf, J. McEachin, N. Liu, N. Russell, L. Unumb, S. Shapiro, D. Khosrowshahi, Concerns about aba-based intervention: An evaluation and recommendations, Journal of autism and developmental disorders 52 (2022) 2838–2853.

[3] G. Gordon, C. Breazeal, Bayesian active learning-based robot tutor for children's word-reading skills, in: Proceedings of the AAAI conference on artificial intelligence, volume 29, 2015.

[4] A. Deshmukh, S. K. Babu, R. Unnikrishnan, S. Ramesh, P. Anitha, R. R. Bhavani, Influencing hand-washing behaviour with a social robot: Hri study with school children in rural india, in: 2019 28th IEEE international conference on robot and human interactive communication (RO-MAN), IEEE, 2019, pp. 1–6.

[5] W.-Y. G. Louie, J. Korneder, I. Abbas, C. Pawluk, A study on an applied behavior analysis-based robot-mediated listening comprehension intervention for asd, Paladyn, Journal of Behavioral Robotics 12 (2020) 31–46.

[6] D. Premack, G. Woodruff, Does the chimpanzee have a theory of mind?, Behavioral and brain sciences 1 (1978) 515–526.

[7] L. Cominelli, D. Mazzei, D. E. De Rossi, Seai: Social emotional artificial intelligence based on damasio's theory of mind, Frontiers in Robotics and AI 5 (2018) 6.

[8] W. Mou, M. Ruocco, D. Zanatto, A. Cangelosi, When would you trust a robot? a study on trust and theory of mind in human-robot interactions, in: 2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN), IEEE, 2020, pp. 956–962.

[9] E. Cipani, K. M. Schock, et al., Functional behavioral assessment, diagnosis, and treatment: A complete system for education and mental health settings, Springer Publishing Company, 2010.

[10] Z. Peng, K. Mo, X. Zhu, J. Chen, Z. Chen, Q. Xu, X. Ma, Understanding user perceptions of robot's delay, voice quality-speed trade-off and gui during conversation, in: Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems, 2020, pp. 1–8.

[11] T. Shiwa, T. Kanda, M. Imai, H. Ishiguro, N. Hagita, How quickly should communication robots respond?, in: Proceedings of the 3rd ACM/IEEE international conference on Human robot interaction, 2008, pp. 153–160.