Automated Action Selection and Embodied Simulation for Socially Assistive Robots using Standardized Interactions

Kim Baraka

CMU-RI-TR-20-38

The Robotics Institute, Carnegie Mellon University, Pittsburgh PA 15213
Instituto Superior Técnico, Universidade de Lisboa, 2744-016 Porto Salvo, Portugal

Submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Robotics

Thesis committee:
Manuela Veloso, (*co-chair*, CMU/JP Morgan AI Research)
Francisco S. Melo (*co-chair*, INESC-ID/IST)
Henny Admoni (RI, CMU)
Aaron Steinfeld (RI, CMU)
Luca Iocchi (University of Rome "La Sapienza")
Iolanda Leite (KTH Royal Institute of Technology)

June 2020

To my cousin Céline, whose physical and mental worldviews continue to fascinate me.

Abstract

Robots have the tremendous potential of assisting people in their lives, allowing them to achieve goals that they would not be able to achieve by themselves. In particular, socially assistive robots provide assistance primarily through social interaction, in healthcare, therapy, and education contexts. Despite their potential, current socially assistive robots still lack robust interactive capabilities to allow them to carry out assistive tasks flexibly and autonomously. Some challenges for these robots include responding to and engaging in multi-modal behavior, operating with minimal expert intervention, and accommodating different user needs.

Motivated by these challenges, this thesis aims at augmenting the algorithmic capabilities of such robots by leveraging the structure of existing standardized human-human interactions in assistive domains. Using therapy for Autism Spectrum Disorder (ASD) as a domain of focus, we explore two roles for a socially assistive robot: 'provider' and 'receiver'.

In the provider role, the robot proactively engages in assistive tasks with a human receiver (namely a child with ASD), following standardized interactive tasks. We contribute a family of algorithms for automated action selection, whose goal is to build cost-optimal robot action sequences that account for a range of receiver profiles. We further estimate the action parameters needed to run these algorithms through empirical studies with children with ASD and psychology experts, and show that the algorithms are able to generate personalized action sequences according to different child profiles.

In the receiver role, the robot simulates common behavioral responses of children with ASD to the standardized actions, acting as an aid for providers in training. By reversing the standardized diagnosis pipeline, we first develop a simulation method that generates behaviors consistent with user-controllable receiver profiles. In a second step, we develop an interactive robot capable of responding to a therapist's actions in an embodied fashion.

Our evaluation studies conducted with therapists validate the designed robot behaviors and show promising results for the integration of such robots in clinical training.

These contributions allow for a richer set of interactions with robots in assistive contexts, and are expected to increase their autonomy, flexibility, and effectiveness when dealing with diverse user populations.

Keywords: Socially Assistive Robotics, Human-Robot Interaction, Autism Spectrum Disorder, Standardized Interactive Tools, Assistive Algorithms.

Acknowledgements

First, I would like to thank my advisors for their personalized guidance and feedback throughout my work, and for trusting me to be independent in my research. Manuela, thank you for taking me in your lab since my early time as a Master's student, and for supporting me all along. Francisco, thank you for giving me the opportunity to be part of this incredible opportunity to pursue a PhD between two continents. This experience has shaped me both professionally and personally.

I express my gratitude to the other members of my thesis committee, Henny Admoni, Aaron Steinfeld, Iolanda Leite, and Luca Iocchi, whose feedback helped significantly improve the quality of the research and presentation. Henny, thank you for your precious support in research and teaching. Aaron, thank you for acting as a true mentor and providing me with support, connections, advice, and opportunities. Iolanda, thank you for your support and openness. Luca, thank you for joining my committee on such short notice and for supporting my work. Finally, thank you to Ana Paiva for her considerable input and feedback on my work and presentations.

This work would have not been possible without the input of my collaborators. I am forever grateful for Marta Couto's involvement in this thesis and for her endless patience. Marta contributed significantly on the study with children with ASD, as well as facilitating my involvement with the Hospital Garcia de Orta. At the hospital, Anabela Farias and Cátia Jesus were of exceptional help and support, and I am grateful for having been able to use the facilities of the Child Development Center. Thanks to undergraduate researchers Harleigh Awner, Jocelyn Huang, Minji Kim, and Patrick Lin for their direct contributions to the simulation work, to Liz Carter for her input on pilots, to Laura Herlant for her early stage feedback on the research, and to members (and ex members) of the CORAL lab for readily providing help.

The research in this thesis was partially supported by national funds through the Portuguese Fundação para a Ciência e a Tecnologia under projects UID/CEC/50021/2019 and UIDB/50021/2020 (INESC-ID multi-annual funding) and the Carnegie Mellon Portugal Program and its Information and Communications Technologies Institute, under project CMUP-ERI/HCI/0051/2013. I acknowledge the PhD grants with references FCT/6907/14/6/2017/S and SFRH/BD/128359/2017 from the Portuguese Fundação para a Ciência e a Tecnologia. Additionally, the data used for the simulation work was obtained from the Hospital Garcia de Orta and from the National Institute of Mental Health Data Archive (NDA) (formerly National Database of Autism Research (NDAR)). The views and conclusions contained in this document are those of the author only.

I am grateful to the families of children who took part in the study and in the pilots (special thanks to Alberto Sardinha and Rui Prada). I am also thankful for the comments of reviewers and to anyone who gave me a suggestion or comment on my talks, at a conference, during a practice session, or over coffee.

Last but not least, I would like to thank friends and family that have supported me throughout this process, Patrícia, Filipa, and Silvia for your spoken and unspoken support, Nina for your aesthetic organization genius. Poli, your emotional and aesthetic input were crucial for the last sprint of this marathon.

Contents

List of Figures **xv**

List of Tables **xvii**

1 Introduction **1**

1.1 Thesis question 4

1.2 Approach 4

1.2.1 Interaction model 4

1.2.2 Automated action selection *(robot-as-provider)* 5

1.2.3 Embodied simulation *(robot-as-receiver)* 6

1.3 Contributions 7

1.4 Reading guide to the thesis 9

2 Child-Robot Interaction Study **11**

2.1 An ADOS-inspired robotic prompting scheme 13

2.1.1 Interaction setup 13

2.1.2 Action scales 15

2.1.3 Robot control 17

2.1.4 Child profile assessment 17

2.1.5 Robot modes 19

2.2 Interaction scenario 20

2.2.1 Storytelling design 21

2.2.2 Interaction timeline 21

2.3 Data collection 23

2.3.1 Participants 23

2.3.2 Experimental procedure . . . 24
2.3.3 Counterbalancing . . . 25
2.4 Results . . . 26
2.4.1 Wizard coding method validation . . . 26
2.4.2 Assessment results . . . 26
2.4.3 Comparison of modes . . . 28
2.4.4 Qualitative observations . . . 30
2.5 Discussion . . . 31
2.5.1 Children profiles . . . 31
2.5.2 Effect of sequencing . . . 32
2.5.3 Cross-task differences . . . 33
2.5.4 Robotic platform . . . 33
2.6 Related work . . . 34
2.6.1 Robotic personalization and adaptation in the Autism Spectrum Disorder (ASD) domain . . . 34
2.6.2 Attention-related robotic tasks . . . 35
2.6.3 Robot-administered diagnosis . . . 35
2.6.4 Robotic storytelling . . . 36
2.7 Summary . . . 37

3 Optimal Action Sequences for Robot as Provider 39
3.1 Assumptions . . . 39
3.2 Chapter goal . . . 40
3.3 Mathematical framework . . . 42
3.3.1 Problem formulation . . . 43
3.3.2 Optimal sequence generation . . . 45
3.3.3 Analysis of optimal solutions . . . 47
3.4 Framework extensions . . . 50
3.4.1 Trial-sensitive case (TS) . . . 51
3.4.2 Cost-sensitive case (CS) . . . 52
3.4.3 Repetition-sensitive case (RS) . . . 53
3.4.4 General history-dependent case (G) . . . 55
3.4.5 Analysis of OAssistMe extensions . . . 55

3.4.6 Simulated example 57
3.5 Application to robot-assisted autism therapy 60
3.5.1 Cost estimation: expert survey 60
3.5.2 Success probability estimation: child-robot interaction data analysis 62
3.6 Discussion 68
3.6.1 Major findings 68
3.6.2 Limitations 70
3.6.3 Applicability to other settings 71
3.7 Related work 72
3.7.1 Probabilistic models for human-agent interaction 72
3.7.2 Intelligent Tutoring Systems (ITS) 73
3.7.3 Healthcare interventions 74
3.8 Summary 74

4 Simulation as Inverse-Assessment 77
4.1 Background and research questions 78
4.1.1 Structure of the Autism Diagnostic Observation Schedule (ADOS) tool 78
4.1.2 Research questions 82
4.2 Feature vector sampling (RQ A) 84
4.2.1 Dataset description 85
4.2.2 Data analysis 86
4.2.3 Descriptor-Based Mean Mapping Sampling (DB-MMS) algorithm 90
4.3 Behavior selection (RQ B) 93
4.3.1 Behavior database creation 93
4.3.2 Identifying task-relevant features 94
4.3.3 Graph representation of behaviors 95
4.3.4 Building the behavior compatibility graphs 96
4.3.5 Graph-based Behavior Selection (GBS) algorithm 98
4.4 ADOS-Sim simulator 99
4.5 Beyond the ADOS 100
4.5.1 Generating new descriptors 100
4.5.2 Applicability to other tools 104

4.6 Related work 105
4.6.1 User simulation for interaction research 105
4.6.2 Simulation of ASD behaviors 106
4.7 Summary 107

5 Interactive Robots for Provider Training 109
5.1 Why robots with 'autism-like' behaviors? 109
5.2 Interaction design 111
5.2.1 Tasks 111
5.2.2 Action scales 111
5.2.3 Robot behaviors 112
5.2.4 Autonomous control architecture 113
5.2.5 Behavior evaluation (video-based study) 116
5.2.6 Interaction and potential benefits assessment ('in situ' study) . . . 118
5.3 Results and discussion 122
5.3.1 Accuracy results 123
5.3.2 Agreement results 125
5.3.3 Order effects 127
5.3.4 Questionnaire results 127
5.3.5 Qualitative observations 129
5.3.6 Additional remarks 129
5.4 Potential impact 130
5.4.1 Therapist training 131
5.4.2 Autism therapy 132
5.4.3 Education and sensitization 133
5.5 Related work 134
5.5.1 Interactive provider training 134
5.5.2 Physically embodied simulation platforms 134
5.6 Summary 135

6 Conclusion and Future Work 137
6.1 Summary of contributions 137
6.2 Future work 143
6.2.1 Robot-as-provider 143

6.2.2 Robot-as-receiver 143
6.3 Broader impact and lessons learned 144
6.4 Concluding thoughts 146

Bibliography **147**

Appendix A Index **161**

Appendix B Proofs and Additional Results **165**
B.1 Proofs 165
B.2 Additional results 169

List of Figures

1.1 Robot-as-provider and robot-as-receiver components of the thesis approach 5
1.2 Overview of chapter contributions 9

2.1 Interaction setup with robot as provider 14
2.2 Robot control architecture 18
2.3 Chronological timeline of the child-robot scenario 22
2.4 Snapshots from the experimental sessions 25
2.5 Distribution of children profiles during interaction with the robot and with a human 27

3.1 Generic example illustrating the concept of optimizing action sequences in relation to given action success probabilities and costs 42
3.2 Worked example 46
3.3 Sample graphical representation of O^*_T versus O^*_{T-1} as a piecewise linear function 49
3.4 Time complexity curves for the different algorithms as a function of horizon and number of actions 57
3.5 Illustrative comparison of optimal action sequences generated by the different versions of OAssistMe 59
3.6 Effect of R on the total probability of failure for the different versions of the algorithm 60
3.7 Success probability results showing data points and fitted surfaces 65
3.8 Effect of R parameter on total probability of failure to inform R value selection 67
3.9 Sequences generated by the OAssistMe algorithm 68

4.1 Simulation as inverse-assessment 83
4.2 2D visualization of the dataset using a Self-Organizing Map (SOM) . . . 87
4.3 Distribution of data points in the 3D space formed by the ADOS subtotals 89
4.4 Spearman correlation matrix of feature values for the dataset 89
4.5 Correlation matrices generated by the mean sampling method 92
4.6 Histogram of total scores for 100,000 sampled feature vectors 92
4.7 Sample behavior compatibility graph 96
4.8 Overview of the ADOS-Sim architecture 99
4.9 Snapshot of the ADOS-Sim user interface 100
4.10 Generation of new data-driven descriptors through K-means clustering visualized in the ADOS subtotal space 102

5.1 Example of an action scale performed by two researchers 112
5.2 Overview of the autonomous robot architecture 115
5.3 Questionnaire structure organized by categories and items within each category 120
5.4 Autism therapist interacting with the robot and coding its responses according to the ADOS specifications 122
5.5 Confusion matrices of responses for each robot behavior across the two studies 125
5.6 Summary of responses to questionnaire items 128
5.7 Robotic platforms with increasing expressive modalities 131

6.1 Summary of the robot-as-provider approach 138
6.2 Summary of the robot-as-receiver approach 139

List of Tables

2.1 Summary of the robot's actions inspired by the actions of ADOS tasks . 16
2.2 Comparison of success occurrences in the three modes 29

3.1 Examples of action scales used in speech therapy and education 41
3.2 Mean estimated costs and standard errors for actions in the two tasks based on experts' responses 62
3.3 Multiple logistic regression results on the child-robot interaction data . . 64
3.4 Minimum R values for acceptable algorithm performance 66

4.1 List of the ADOS standardized tasks 79
4.2 List of the ADOS features used for the scoring algorithm 81
4.3 Features identified as relevant for each task based on the ADOS manual . 95

5.1 Summary of the defined action scales the human can use to interact with the robot 112
5.2 Summary of the designed 'autism-like' robot behaviors of varying severities 114
5.3 Accuracy results per participant across the two studies 123
5.4 Accuracy and agreement results per feature 123
5.5 Percent agreement across participant responses, in comparison with values from the ADOS literature 126

A.1 List of main acronyms used in this thesis 161
A.2 Guide to the notation used in this thesis. Bolded variables represent vectors or sets, and starred variables represent optimal quantities. 162
A.3 Full list of features from the ADOS Module 1 163

B.1 Action sequences generated by OAssistMe extensions (data from mode Explore) 169
B.2 Action sequences generated by OAssistMe extensions (data from modes Explore + Therapy) 170
B.3 Action sequences generated by OAssistMe extensions (data from modes Explore + Assess) 170
B.4 Action sequences generated by OAssistMe extensions (data from modes Explore + Therapy + Assess) 171

List of Algorithms

1 OAssistMe: Linear-time algorithm to find an optimal action sequence for horizon T 46
2 TS-OAssistMe: Trial-sensitive extension of OAssistMe where success probabilities are a function of trial 52
3 CS-OAssistMe: Cost-sensitive extension of OAssistMe where success probabilities are a function of cost of history 53
4 RS-OAssistMe: Repetition-sensitive extension of OAssistMe where success probabilities are a function of the number of action repetitions 54
5 Pseudocode of the Descriptor-Based Mean Mapping Sampling (DB-MMS) algorithm 93
6 Pseudocode of the Graph-based Behavior Selection (GBS) algorithm . . . 98

Chapter 1

Introduction

Robots have the tremendous potential of assisting people in their lives, allowing them to achieve goals that they would not be able to achieve by themselves. In particular, *socially assistive robots* are robots that assist people primarily through social interaction to achieve progress over time [58]. Such robots have been developed in the context of therapy and rehabilitation [133, 140, 30], elderly care [60, 8, 41], assistance to people with sensory impairments [27, 111, 35], and education [23, 74, 96], among others.

Despite their potential, current socially assistive robots still lack robust interactive capabilities to allow them to carry out assistive tasks flexibly and autonomously. The challenges associated with such a goal are numerous. They include the ability to respond to and engage in *multi-modal behaviors* that integrate verbal and non-verbal social cues (speech, gaze, gestures, lights, etc.), the ability to *operate autonomously* or with minimal human guidance, and the ability to *personalize* their interactions for users with different needs or preferences.

This thesis aims at augmenting the algorithmic capabilities of socially assistive robots by leveraging the structure of existing *standardized tools for human-human interaction*. Such tools include diagnostic tests, assessment scales, protocols for intervention, all of which are widely used in healthcare and education fields. These tools allow a *'provider'* (e.g., therapist, teacher) to efficiently interact with a *'receiver'* (e.g., patient, student) according to standardized tasks. Our research explores both of these roles for the robot. In the provider role, the robot provides assistance to a human receiver, informed by the procedures of the interactive tool. In the receiver role, the robot is used to simulate a standardized interaction with receivers of varying profiles, thereby assisting the providers in their training.

To pursue our research goals, we commit to go in depth into one application domain, namely therapy for children with Autism Spectrum Disorder (ASD)[1]. In this context, the provider is a therapist and the receiver is a child with ASD. The main standardized tool utilized throughout the thesis is the Autism Diagnostic Observation Schedule (ADOS) [101], which is considered the gold standard for ASD diagnosis. It specifies *standardized tasks* that dictate how therapists should interact with children when administering the tool, as well as a *coding scheme* that maps observed behaviors to values on features spanning several behavioral dimensions (specifics of the ADOS can be found in Chapter 4). The robot used throughout the thesis is the NAO robot[2], a humanoid robot with arms, legs, and a head, expressive lights, and sensors including a camera, microphone arrays, and touch sensors. The NAO robot has been used in a large number of studies with children with ASD and has had general positive response in terms of its embodiment and motion behavior [4, 146, 70, 55, 153].

Despite our focus on a single application domain and robotic platform, most of the contributions of this thesis lie at a level of abstraction that allows them to be easily applied to other assistive domains and robots. Therefore every chapter will include some form of discussion about potential generalization beyond our domain of focus.

Why robots and autism?

ASD is a set of developmental disorders that affects social abilities, verbal and non-verbal communication, and potentially motor and cognitive skills [5]. According to a 2016 report by the Centers for Disease Control and Prevention [9], one in 54 children in the US has some form of ASD. In past years, the introduction of robots in therapy for children with ASD has gained a lot of interest [132, 31, 46, 149, 51]. Socially assistive robots offer a number of characteristics that make them attractive tools for use in autism therapy, including:

[1]There is a debate on whether to use disability-first (‘autistic person’) versus person-first language (‘person with autism’), as different people have different preferences [85]. In this thesis, we stick to person-first, because it reminds the reader that a person with a disability is more than the ‘set of features’ that our work uses to model them.

[2]https://www.softbankrobotics.com/emea/en/nao

- *Predictability* — They are more predictable than humans, with programmed behaviors that are generally mechanistic, repeatable, and triggered by particular environmental conditions. While the unpredictability associated with a human interaction can cause immense distress in individuals with ASD, with a robot children can explore interactions with less social anxiety.
- *Social simplicity* — They are able to engage in multimodal social interactions, focusing on social communication aspects that are major impairments of ASD. They do so in a way that is simplified and reproducible, making it easier for children to process the information, and potentially enabling them to generalize what they learn with the robot to more complex interactions involving humans.
- *Control* — They allow for higher control of therapy methods, as well as objective data gathering for monitoring children's progress and complementing therapists' subjective evaluation.

Furthermore, in relation to our research goals, the autism domain is a particularly interesting and challenging one to explore in socially assistive robotics for the following reasons:

- *High-variance population* — Individuals with ASD are characterized by their diversity of profiles (hence the term 'spectrum'), which motivates the need for robust personalization mechanisms within the context of robot-assisted therapy. As renowned autism researcher Dr. Stephen Shore puts it: "If you've met one person with autism, you've met one person with autism."
- *Importance of social interaction* — Social and communication abilities are the core areas of deficits in ASD, and the focus of most ASD therapies. The importance of this socially interactive component prompts us to consider types of interaction with robots that incorporate multiple modalities including verbal and non-verbal communication such as gaze, gestures, touch, visual cues, and sound.
- *Interactive diagnosis* — Unlike a number of other disorders, autism is diagnosed based on interaction and behavioral observation. There is no blood test or imaging procedure that can accurately detect the presence or severity of the disorder. The interactive diagnostic tools used for autism are therefore a rich source of structure and data to inform intelligent social behavior of robots operating in this domain.

Motivated by these aspects, we now state the overarching research question for this thesis.

1.1 Thesis question

This thesis seeks to answer the following question:

> ***How can standardized interactive tools be used to expand the socially assistive capabilities of robots?***

We approach this question from two perspectives: *robot-as-provider* and *robot-as-receiver*. In the robot-as-provider component of the approach, where the robot assists the receiver directly, we use the ADOS tool directly to inform how a robot interacts with children with ASD of different profiles in the context of therapeutic tasks. We specifically contribute a general algorithmic framework to allow for personalized *automated action selection* on the robot. In the robot-as-receiver component of the approach, we contribute a reversal of the interaction model using computational methods, in order to achieve an *embodied simulation* of some common behavioral responses for a range of different receivers. Such a robot can be used to assist providers in the context of simulated training. Figure 1.1 gives a high-level summary of our approach, discussed in more details in the next section.

1.2 Approach

We start by briefly presenting the interaction model used in this thesis, then outline both components of our approach in more detail.

1.2.1 Interaction model

A typical interaction between provider and receiver happens according to standardized *tasks*, which we assume can be used either for assessment (e.g., diagnosis) or intervention (e.g., therapy). Every task is a procedure to be followed, consisting of standardized *actions* that the provider selects according to the purpose of the interaction (e.g., assessment, intervention). The action selection method is also a function of the *receiver profile*. The

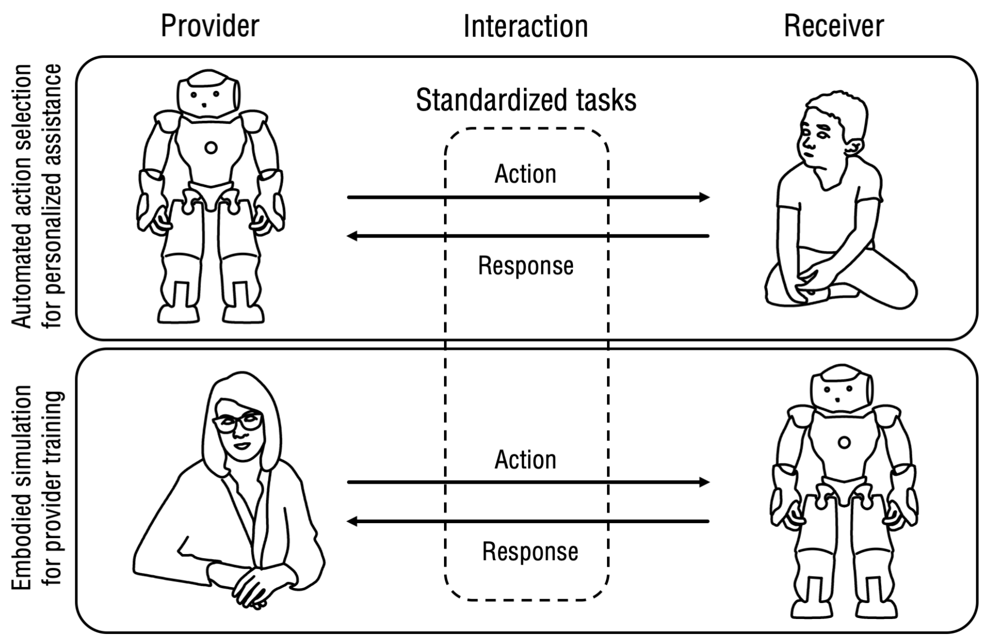

Figure 1.1 Robot-as-provider and robot-as-receiver components of the thesis approach (top and bottom respectively). Standardized tasks are assumed to be similar across the two components, and are modeled as sequences of provider actions and receiver responses.

latter models receivers through a set of coded *features* that characterize how receivers behave in the context of an interaction with the provider. Receiver *behaviors* are understood as behavioral responses to the provider's actions or tasks. In the context of assessment, the provider has the ability to observe these behaviors and estimate the receiver profile accordingly. These interaction components appear in Figure 1.2, included at the end of this chapter.

1.2.2 Automated action selection *(robot-as-provider)*

In a first step, we conduct an exploratory study to measure how children with different severities of ASD respond to actions of a NAO in two tasks related to attention mechanisms using video screens. We integrate several instances of the tasks in a storytelling scenario featuring cartoon snippets to keep the children engaged. The robot first assesses the profile of the child, then alternates between a random action selection and a therapist-inspired action selection. This study provides us with data to quantify the behavioral responses of children in the tasks with the robot, and is represented by the box 'Interaction study data collection' in Figure 1.2.

In a second step, we formalize the problem of personalization as a general optimization problem in which actions have (therapeutic) costs and success probabilities. We specifically consider tasks where the robot needs to plan a sequence of actions over a fixed time horizon, with the goal of achieving a success with the lowest expected cost possible. We contribute *OAssistMe* (shown in Figure 1.2), a dynamic programming algorithm that generates cost-optimal action sequences given the action parameters, and investigate several extensions of it, motivated by different application domains. We provide a thorough analysis of the algorithms, including proofs for a number of properties of optimal solutions that we show align with typical human provider strategies.

Finally, we instantiate our theoretical framework in the context of the robot-assisted therapy tasks considered in our study. In this context, we present methods for determining action parameters based on the data from the study (to determine action success probabilities) as well as an expert survey (to determine action costs). We show that the algorithm is able to generate different action sequences for different receiver profiles and different tasks. In relation to our research question, this component of the approach specifically addresses the lack of personalization and adaptation strategies of current socially assistive robots.

1.2.3 Embodied simulation *(robot-as-receiver)*

We develop *ADOS-Sim* (shown in Figure 1.2), a simulator of behavioral responses commonly seen in children with ASD in the context of the standardized ADOS tasks. The approach taken stems for the observation that such simulation can be seen as an inverse-assessment operation. While assessment maps behaviors to a receiver profile, simulation takes a profile and generates behaviors. By reversing the chain of the ADOS, we depart from high-level descriptors of a child's profile, such as ASD severity, age, and language ability, to individual realistic behaviors in the different ADOS tasks. The simulator is mainly based on two algorithms. The first one, *Descriptor-Based Mean Mapping Sampling (DB-MMS)*, generates synthetic profiles, represented as feature vectors, from a measure of ASD severity. It is informed by a dataset of real ADOS scores from different sources, collected on children with different ASD severities. The second algorithm, *Graph-based Behavior Selection (GBS)*, selects behaviors within tasks, while ensuring that no conflict occurs between any pair of behaviors.

Building upon ADOS-Sim, we enable the robot to exhibit 'autism-like' behaviors with controllable degrees of severity along several features (represented by the box 'Embodied simulator' in Figure 1.2). We first design 16 robot behaviors spanning four different ADOS features, namely ones related to response to join attention, response to name calling, pointing, and language ability. We then integrate our designed behaviors into an autonomous control architecture. The robot is capable of having continuous interactions with one or more humans, according to the pre-defined actions it recognizes. It can be customized by specifying an arbitrary severity for each feature, resulting in 256 unique combinations. We evaluated, in both video-based and 'in situ' studies, the validity of the designed behaviors and the potential of our approach for complementing therapist training. In relation to our research question, this component of the approach introduces a novel way for robots to assist providers as embodied simulators used for training.

1.3 Contributions

Our main contributions can be grouped into three categories: algorithmic, methodological, and autism-related.

Algorithmic contributions

- ADOS-Sim, a simulator that outputs behaviors consistent with high-level children profiles. The two main components of the simulator are the DB-MMS algorithm, for data-driven feature generation, and the GBS algorithm, for conflict-aware behavior selection.

- OAssistMe, a linear-time algorithm that generates optimal action sequences given action costs and success probabilities. We also provide proofs for properties of its optimal solutions.

- Three extensions of the algorithm (trial-sensitive, cost-sensitive, and repetition-sensitive) that add different assumptions of dependency on the history of actions.

Methodological contributions

- A methodology for simulation as inverse-assessment, illustrated using the structure of the ADOS diagnostic tool in addition to a database of ADOS scores.
- A methodology for the use of robots as receivers to assist the training of providers, preliminarily evaluated in video-based and ‘in situ’ studies.
- A methodology for determining action costs and success probabilities in the context of robot-assisted therapeutic tasks.

Autism-related contributions

- An analysis of ADOS data from different sources using dimensionality reduction techniques.
- A scenario based on interactive storytelling, integrating ADOS-inspired tasks that address specific autism impairments related to attention deficits.
- A preliminary computational model of child response to the robot’s actions in the context of ADOS-inspired tasks.

Figure 1.2 summarizes how these contributions fit within the chapter structure of the thesis, in relation to the thesis concepts previously introduced.

Taken together, these contributions allow for a richer set of interactions with robots in assistive contexts, and are expected to increase their autonomy, flexibility, and effectiveness when dealing with populations characterized by diverse profiles, needs and preferences.

Disclaimer: Although an important aspect to consider for technology adoption, the demonstration of the (long-term) benefits for any of the contributions made in this thesis on intended users, both children and therapists, is not in the scope of this thesis. That applies to algorithms, methodology, and scenarios presented in this document. Whenever present, the use of human participants studies is viewed simply as a validation or preliminary evaluation tool for the technological questions studied.

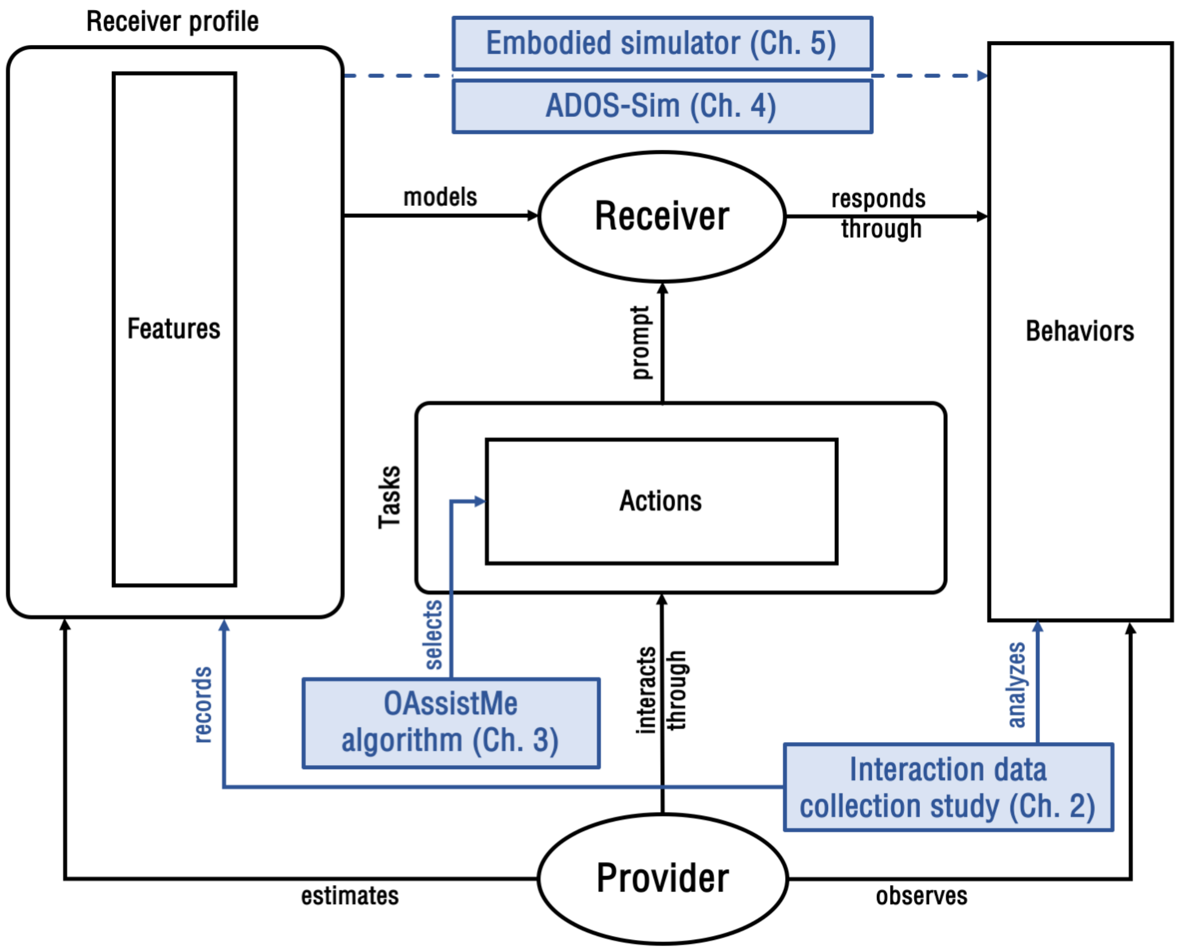

Figure 1.2 Overview of chapter contributions (in blue) in relation to the core components of the interaction model.

1.4 Reading guide to the thesis

The outline below describes the content of each chapter, grouping related and complementary contributions together. Every chapter ends with a section describing related work for that chapter, followed by a summary of the chapter's contributions and main results.

Chapter 2: Child-Robot Interaction Study — We design and run an exploratory study based on interactive storytelling with structured tasks to collect data on child-robot interaction, in order to inform methods of personalized action selection.

Chapter 3: Optimal Action Sequences for Robot as Provider — We formalize the personalization problem as an optimization problem that takes into account action costs and success probabilities. We present different versions of OAssistMe, an algorithm that generates optimal action sequences building on the data collected in Chapter 2.

Chapter 4: Simulation as Inverse-Assessment — We present ADOS-Sim, a simulator that outputs behaviors consistent with different children profiles, in the context of standardized ADOS tasks.

Chapter 5: Interactive Robots for Provider Training — We describe how we extend our simulation approach to visualize behaviors in an embodied way on a robot. We present results from two studies where therapists evaluate the resulting embodied simulator.

Chapter 6: Conclusion and Future Work — We summarize our contributions and end by describing avenues for future work that builds on this thesis.

Appendix A: Index — We provide tables of acronyms, notation, and ADOS information used throughout the thesis for easy referencing.

Appendix B: Proofs and Additional Results — We present mathematical proofs as well as additional results for the algorithms of Chapter 3.

Chapter 2

Child-Robot Interaction Study

This chapter considers the robot in the provider role, specifically in the context of a robot-assisted autism therapy scenario targeting attention skills, a major area of impairment for young children with ASD. We report on an exploratory study whereby a NAO humanoid robot engages with 11 children with different ASD severities in a storytelling scenario integrating structured ADOS-inspired tasks. The study, run in collaboration with a psychologist and therapists from a child development center at a Portuguese hospital, aimed at analyzing the role of action sequencing on children's response to the robot's actions. While the actions available to the robot within the structured tasks are pre-defined, we controlled the *action sequences* that the robot executed within different instances of the same task. In particular, we considered three modes of operation for the robot, corresponding to three different ways of generating action sequences. The first mode, *Assess*, is inspired by the ADOS procedure within the tasks of interest. The second mode, *Therapy*, uses the profile assessed in the previous mode to generate action sequences inspired by the way therapists would select their actions in alignment with therapeutic goals. The third mode, *Explore*, generates completely random action sequences.

This exploratory study will help us inform the design of algorithms for personalization and adaptation of the robot's action selection in the next chapter. Personalization and adaptation are widely used strategies amongst autism therapists, but the challenges of achieving such mechanisms in robot-assisted therapeutic tasks are numerous. These challenges include:

- *Assessment* — Building useful profiles of children interacting with robots consists in assessing features characterizing their interaction with the robot. This is a challenging goal for the following reasons:

  - Child response to robots may significantly differ from response to humans, which means there might not be a systematic way to predict response to a robot given data on interaction with a human.
  - The cost of exploration may be high. Individuals with ASD are often extremely sensitive to details, and a single 'wrong step' in the robot's behavior may result in serious consequences, such as jeopardizing the willingness of the child to interact again with the robot.
  - The amount of data that a robot can collect with a specific child is limited, which makes it difficult to estimate, from scarce data, child features that are useful for the interaction.

  In our work, we base our feature assessment method on standard diagnostic procedures widely used by human therapists.

- *Personalization* — Personalizing robot behavior to each receiver profile is another research question that requires domain knowledge. What strategy works best for which profile? How can its efficacy be measured?

  In this chapter, our personalization strategy in mode Therapy aligns with typical strategies followed by human therapists that have been shown to promote learning in the long-term.

- *Integration in naturalistic context* — Since most ASD therapy tasks rely on aspects of social interaction, it is necessary to integrate them in an engaging scenario with a consistent context and progression. Maintaining stable engagement levels with such a population is particularly challenging and also particularly helpful as it reduces uncertainty in the robot's ability to predict children's responses.

  In our study, we integrate structured tasks of interest within a larger interactive storytelling scenario.

In a first step, we leverage the structure of the ADOS tool to develop a set of prompting actions on a NAO humanoid robot (Section 2.1). These actions are aimed at eliciting a goal response from the child in two attention-related tasks. The robot actions (‘presses’ in ADOS terminology) fall under a scale organized by increasing levels of explicitness, adapted to a range of child profiles. Based on these robotic actions, we develop a control architecture that allows the robot to prompt the child with different sequences of actions according to its mode of operation – Assess, Therapy, or Explore. We integrate the structured tasks in an interactive storytelling scenario involving the robot and two controllable screens showing story-related cartoon excerpts (Section 2.2). We then collect and analyze data on the behavioral responses of 11 children with ASD during a session with the robot (Section 2.3). In addition to providing insight on how action sequencing affects child response, the data will be used in the next chapter to build a probabilistic profile-dependent model of child response.

2.1 An ADOS-inspired robotic prompting scheme

In this section, we describe our robotic prompting scheme developed for a NAO humanoid robot, and inspired by the ‘algorithmic’ nature of two ADOS tasks, related to joint attention and response to name. After describing the interaction setup considered, we present our developed robotic actions inspired by these ADOS tasks. We then discuss our flexible robot control architecture, which allows for different modes of operation (namely Assess, Therapy, and Explore).

2.1.1 Interaction setup

Figure 2.1 shows the physical setup used in this chapter, inspired by the work of Warren et al. (2015) who demonstrated its suitability for young children with ASD [152]. We found this scenario to be attractive to explore the idea of personalization of attention-related interactions, as it allows for both control and flexibility when compared to scenarios involving physical objects, portable digital devices (e.g., tablets) [24], or scenarios where the child moves around the space [107]. The setup consists of a NAO robot standing on a table, at which the child is seated, and two 49.4 cm LCD screens positioned at around a 90 degree angle on both sides of the child’s chair.

The robot engages in two main tasks of focus, inspired by the ADOS:

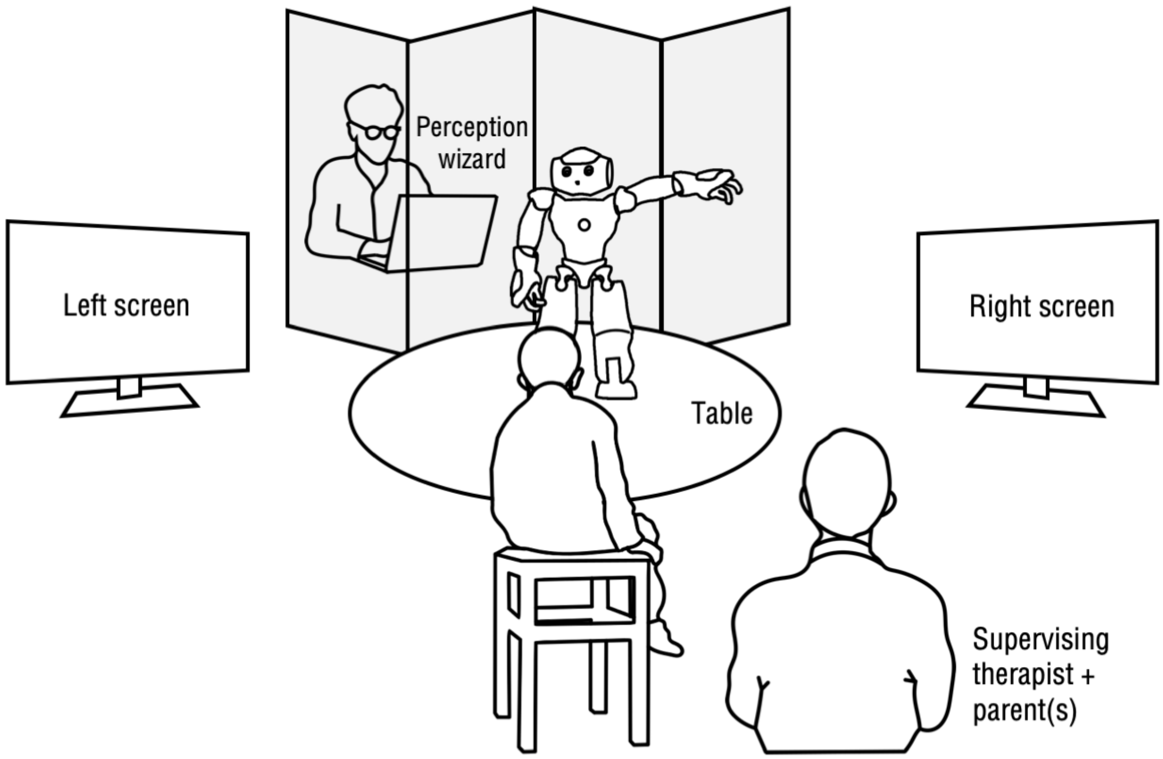

Figure 2.1 Interaction setup with robot as provider. Figure is only meant for illustrative purposes; relative positions and sizes of the components are not exact.

- **'Joint Attention' task (JATT)** — The robot directs the child's gaze from looking at the robot to looking at a target screen where a video will play.
- **'Name Calling' task (NAME)** — The robot directs the child's gaze from looking at the video on one of the screen back to looking at the robot.

A 'perception Wizard' provides the robot with information about the child's gaze behavior through a computer interface, hidden behind a single-sided mirror at an angle that maximizes the view to the scene. Specifically, during each of the two tasks, the Wizard is responsible for triggering a 'success' event whenever the child performs the goal behavior for that task (i.e., orienting their gaze in the right direction). For the JATT task, a success triggers a short video snippet. For the NAME task, a success stops the video playing on the screen where the child is looking. While eye-tracking or head-tracking technology were available for us to use, we preferred to rely on human perception, as such technologies are too invasive and inaccurate, especially for children with attention impairments who tend to move considerably. Furthermore, it allows us to focus on the action selection problem, while factoring out the additional noise that comes with an automated perception system.

A single processing unit allows the control of each screen individually. The Wizard's machine runs the main software to automatically control the behavior of both the robot and the screens, while allowing the Wizard to provide success information when needed. A wired network connection through a switch between all computing units was used to minimize delays and connectivity issues. We used the Thalamus framework [124] to facilitate communication between the distributed modules.

For safety purposes, the robot's feet were stuck to the table using tape to avoid falls, as we have noticed that some children were particularly keen on touching and poking the robot. Next we describe the actions that we programmed the robot to execute during the two tasks.

2.1.2 Action scales

As part of the ADOS tasks, there exist systematic 'algorithms' for evaluating a child's response to joint attention and response to name, through scales of actions with increasing levels of explicitness ('hierarchies of presses' in ADOS terminology). Each action corresponds to a more or less explicit action taken by the therapist with the common aim of eliciting a goal behavior on the child's part. The ADOS actions and the goal child behaviors are summarized in columns 2–4 of Table 2.1.

Inspired by the structure of the ADOS tasks, we developed similar action scales for the robot, aiming to elicit the corresponding goal behavior from the child. Column 5 of the table summarizes our developed robotic actions. We should point out that the aim was not to replicate the content of the ADOS actions with high fidelity. Rather, we came up with similar scales adapted to our scenario and accounting for a range of responses along the scales. Also, to ensure an increasing level of explicitness for the actions, we structured them such that action $a + 1$ is a replica of action a with an added element that either adds intensity to the stimulus (e.g., sound on top of video) or facilitates the understanding of the action (e.g., pointing added to gaze). We used the SERA software architecture [123] to control the robot's multi-modal behaviors. Speech was automatically generated by NAO's built-in text-to-speech engine.

Table 2.1 Summary of our robotic actions, organized along a scale with increasing levels of explicitness (1–4), inspired by the actions of ADOS tasks.

Task	**Goal behavior**	**Level**	**ADOS action**	**Robot action**
JATT	Look at target	1	Shift gaze from child to target object + “[Name], look!”	**(Gaze + speech)** Shift gaze from child to target screen + “[Name], look!” (static picture on both screens)
		2	Shift gaze + “[Name], look at that!”	**(Gaze + speech + point)** Shift gaze + “[Name], look at that!” + point (static picture on both screens)
		3	Shift gaze + “[Name], look at that!” + point	**(Gaze + speech + point + video)** Shift gaze + “[Name], look at that!” + point + play muted video on target screen (static picture on other screen)
		4	Activate target (toy)	**(Gaze + speech + point + video + sound)** Shift gaze + “[Name], look at that!” + point + play video with localized sound on target screen (static picture on other screen)
NAME	Look at provider	1	“[Name!]”	“[Name!]”
		2	Ask parent/caregiver to call name	“[Name], look over here!”
		3	Ask parent/caregiver to make a familiar sound	“[Name], look over here!” + blink lights
		4	Ask parent/caregiver to do whatever necessary to get child’s attention	“[Name], look over here!” + blink lights + wave arm

We fine-tuned our actions based on pilot trials with four Typically Developing (TD) children, two children with ASD, and one child with minimal ASD. Specifically, for task JATT, we had to take special care with the behavior of the screens, as it seemed from our pilots that the sharp transitions from a black screen to an image or video was a very salient stimulus that transiently overpowered the robot's role. For this reason, we decided to pre-load a static picture on both screens, corresponding to the first frame of the video to be shown, and to keep the brightness of the screens on a low setting.

2.1.3 Robot control

Figure 2.2 shows the relation between the different modules of the robot control architecture. Before starting the execution of the task, the robot first generates an *action sequence* $\mathbf{\Pi} = \langle a_1, a_2 \ldots, a_T \rangle$, i.e., a plan of actions to be executed over consecutive time steps. An action sequence generation module produces these sequences according to parameters communicated by a high-level decision maker, including task type, robot operation mode (see Section 2.1.5), as well as other scenario- and child-related parameters. The action sequence $\mathbf{\Pi} = \langle 1, 3, 2, 2 \rangle$, for instance, means that the robot will perform action of level 1 as a first trial, then potentially execute more trials with actions of level 3, 2, and again 2, until the goal behavior is observed or the sequence is exhausted. While in this work we restrict the action sequence length T to 4, our architecture is general enough to allow for arbitrary sequences of any length. An action sequence execution module executes the actions on the robot sequentially, until either a success is triggered by the Wizard or the sequence is exhausted. The trigger of the next action in the sequence is a timeout in case no success occurs. Based on our pilots, we set the duration of the timeout to 3.5 seconds.

2.1.4 Child profile assessment

In the ADOS, the therapist goes through the actions hierarchically from least to most explicit until the expected response is observed, and records the level of the first successful action. This number can be seen as a measure of abnormality of response to the task. In this work, since we consider two tasks, the child profile is represented as a pair of features (RJA,RNA), where RJA/RNA is the lowest action level at which a success is observed in task JATT/NAME respectively. If none of the four action levels achieve a success, we assign to the corresponding feature a value of 5. In a typical ADOS session, features

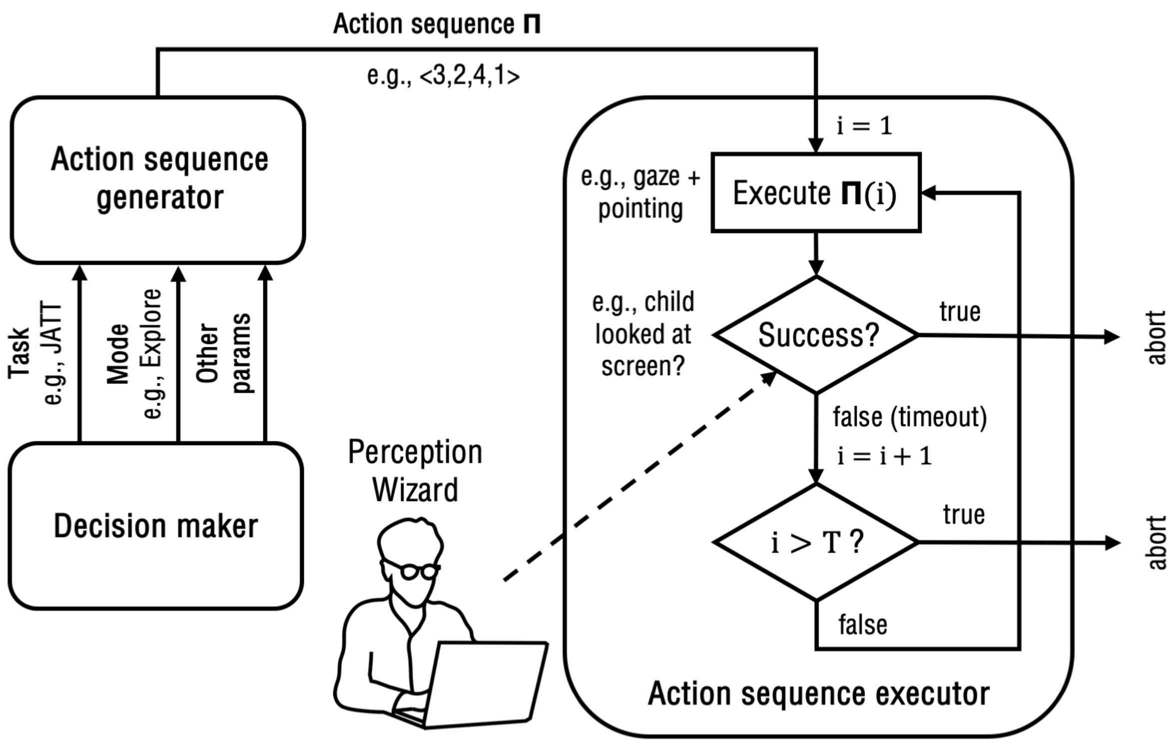

Figure 2.2 Robot control architecture.

RJA and RNA are measured only once. In a robotic scenario however, we expect much greater variability in the response due to the novelty effect associated with the robot, as well as the scenario as a whole. For this reason, accurately estimating values of a feature of interest f may require several measurements. Given n measurements $f^{(1)}, f^{(2)} \ldots, f^{(n)}$, we estimate f as:

$$f = \begin{cases} \mathrm{rnd}(\sum_{1}^{n} \frac{f^{(i)}}{n}) & \text{if } \sum_{1}^{n} \frac{f^{(i)}}{n} \mod 2 \neq 0.5 \\ \mathrm{rnd}(\sum_{2}^{n} \frac{f^{(i)}}{n-1}) & \text{if } \sum_{1}^{n} \frac{f^{(i)}}{n} \mod 2 = 0.5 \end{cases} \tag{2.1}$$

where rnd() represents rounding to the nearest integer. In other words, in case of an estimate lying exactly in the middle of two levels, we omit the first sample. The latter is more prone to novelty factors and is hence, in comparison to more recent samples, less reflective of subsequent performance of the child on the task. Equation 2.1 applies for estimating both RJA and RNA.

Examples ($n = 4$):
$f^{(1)} = 3, f^{(2)} = 3, f^{(3)} = 4, f^{(4)} = 2 \rightarrow f = \mathrm{rnd}(\frac{3+3+4+2}{4}) = 3$
$f^{(1)} = 3, f^{(2)} = 3, f^{(3)} = 2, f^{(4)} = 2 \rightarrow f = \mathrm{rnd}(\frac{3+2+2}{3}) = 2$

2.1.5 Robot modes

We consider three modes of operation for the robot during task execution. These modes effectively translate into different action sequences, as follows:

- **Assess mode** — The robot follows the action scale hierarchically, from least to most explicit action, as is done in the original ADOS tasks. The action sequences for this mode are fixed for all children, and of the form $\mathbf{\Pi} = \langle 1, 2, 3, 4 \rangle$. This mode enables the robot to build a profile of the child by recording the lowest action level at which the child responds for the two tasks, as explained in the previous subsection.

- **Therapy mode** — The robot follows a therapy-inspired action sequence characterized by consistency, repetition, and personalization. For a given child feature f, the first two actions in the action sequence are repetitions of action f, while the last two actions are repetitions of action $f + 1$. In the edge cases where $f = 4$ or $f = 5$, this mode generates four repetitions of action 4.

  Examples:
  $f = 2 \rightarrow \mathbf{\Pi} = \langle 2, 2, 3, 3 \rangle$
  $f = 3 \rightarrow \mathbf{\Pi} = \langle 3, 3, 4, 4 \rangle$
  $f = 4 \rightarrow \mathbf{\Pi} = \langle 4, 4, 4, 4 \rangle$.

  This mode was developed in accordance with typical therapeutic strategies, based on the concepts of 'just-right challenge' and task grading [134, 75], as well as a discussion with autism experts.

  It is important to mention that the goal of this mode is not to minimize the number of actions needed to observe a success, otherwise the robot could always select the most explicit action 4. Instead, in alignment with therapeutic goals, this mode chooses the least explicit action that has been shown to work on a particular child, while adapting the level to a higher one if no success is observed after the exhaustion of half the sequence. This choice promotes learning (in the long term) at the cost of potentially increasing the number of actions needed for a success to occur. This trade-off will be formalized in the next chapter.

- **Explore mode** — The robot follows completely random action sequences, where actions are drawn uniformly and independently at every time step. These action sequences are characterized by inconsistency, unpredictability, and lack of personalization, and therefore have little therapeutic value.

We should point out that in any of the modes presented above, the action sequence represents a plan, whose execution is aborted if a success occurs, i.e., if the child performs the goal behavior. While our robot control architecture allows for more modes than the ones above, those were the ones that best fit our scenario and research goals.

In our child-robot interaction study, described next, the robot first runs in the Assess mode, collecting samples to estimate the child's profile. In a second phase, the robot alternates between the Explore and Therapy modes. While in the future one may consider an algorithm that alternates between an Explore phase and an 'Exploit' phase, the Therapy mode in this chapter does not update its action sequence as a function of the outcome of mode Explore.

2.2 Interaction scenario

In order to test our robotic prompting scheme in the context of an extended social interaction, we designed and implemented an interactive storytelling scenario, in which short excerpts of an animated cartoon on the screens regularly support and illustrate the robot's speech delivery. The JATT task is repeatedly used throughout the interaction to direct the child's attention to one of the two screens where the cartoon excerpt is to be shown. Following this task, the robot uses the NAME task to call back the child's attention and resume the storytelling.

2.2.1 Storytelling design

The story we chose is based on an episode of a Japanese cartoon, Ox Tales, dubbed in European Portuguese. Popular in the previous generation, this amusing cartoon is much lesser known by the younger generation. This reduces the chances of current children having strong (positive or negative) feelings about it. The episode was selected based on the simplicity of the plot and the presence of simple actions for the child to imitate, which the robot uses to engage the child throughout the story. We transcribed, simplified and rewrote the video episode in a storytelling style with simple language to ensure that children with different language abilities would be able to follow the story. We then edited and adapted the length and organization of the story based on our pilot trials, aiming at optimizing for child engagement, clarity of robot speech, and plot simplicity. In parallel to the verbal content of the story, we extracted and edited 12 cartoon snippets of 12 seconds each. We handpicked snippets that showed interesting actions throughout the story, including four snippets whose aim is to introduce specific characters of the story.

The robot used its built-in European Portuguese text-to-speech engine for both the storytelling part and the interactive tasks. Even though pre-recorded voice could have been more engaging and natural-feeling for the sake of storytelling, the choice of text-to-speech aligned with our long-term goal of a personalized and adaptive solution that includes modulating speech content automatically. As a result, we opted for the greatest level of reliable autonomy possible on the robot side. To increase the expressivity of the robot during storytelling, we animated it with a 'Breathing' behavior consisting of swinging its weight side to side between each leg at a rate of 30 times per min. We also added expressive hand gestures, randomly alternating between left and right, inspired by simple gestures typically used by storytellers.

2.2.2 Interaction timeline

Figure 2.3 shows the timeline of the interaction. The scenario alternated between storytelling and interactive prompting using the two tasks. The robot also used imitation prompts meant to keep the child engaged (imitation ability is also commonly impaired in children with ASD and has been the focus of some robot-assisted interventions [156, 145]). The robot started with some greetings, which consisted in introducing itself and asking for the child's name until the child responded (or the parent, in case of failure). The

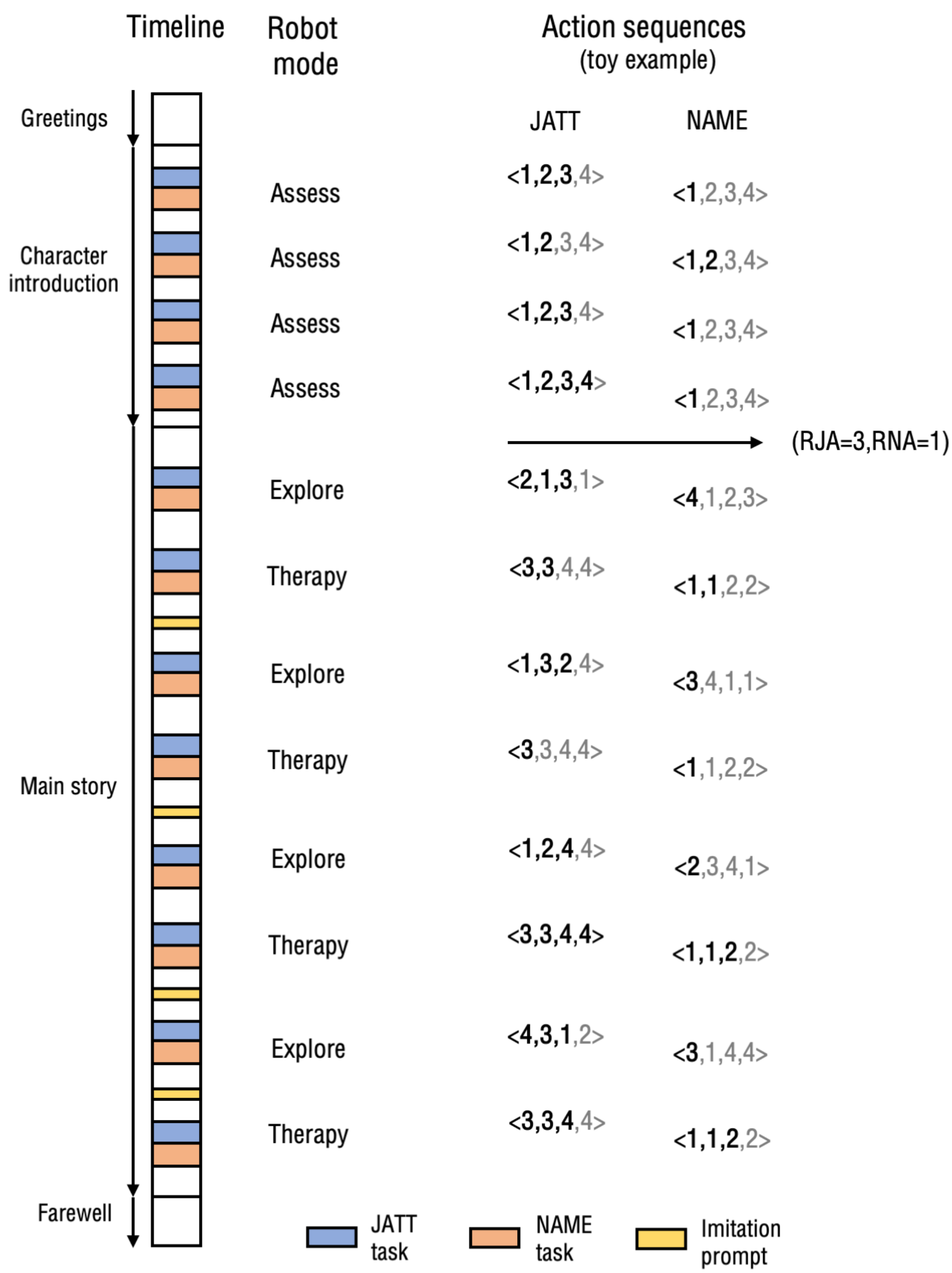

Figure 2.3 Chronological scenario timeline (to approximate scale) along with corresponding robot modes, illustrated with a toy example. Greyed out portions of action sequences represent planned actions that were not executed due to a success.

robot then moved to the assessment phase, in which it presented four characters of the story, using four instances of each task. After the assessment phase, the robot started the main story phase. In both phases, the robot used the cartoon snippets in the tasks to illustrate relevant story content. We tried to balance the number of words as much as possible between the different story parts defined by the occurrence of the tasks. Any success or timeout in JATT triggered the 12-second video snippet of the corresponding part of the story. Any success or timeout in NAME turned both screens to black for a short period of time, then updated both screens with a new static image corresponding to the next part of the story as the robot resumed its speech. In the imitation prompts, the robot asked the child to imitate a total of four gestures related to the story plot (pretending to fly, pretending to run, covering eyes, and looking around). To further keep the child engaged, throughout the story the robot relied on questions such as "What do you think will happen?", or "Will Ox Tales be able to fly?". Right before the main story phase, as well as during the farewell phase, the cartoon theme was played with music on both screens.

In the assessment phase, the robot was always in Assess mode, and performed each task a total of four times. It used the recorded levels at which the children responded to estimate their profile. In the main interaction phase, the robot alternated between the Therapy and Explore modes, performing each task a total of eight times. We remind the reader that the Therapy mode only relied on the result from the assessment phase, and unlike some existing machine learning algorithms that interleave exploration and exploitation in their policies, it was not influenced by the results of the Explore mode.

2.3 Data collection

We collected data on how children with ASD responded to the robot's actions in the scenario described in the previous section. The present section provides details about the study participants and experimental procedure.

2.3.1 Participants

We recruited 11 children with different ASD severities from the Child Development Center at the Garcia de Orta Hospital in Almada, Portugal, to participate in the study. These children did not include the seven participants of our pilot trials. The criteria for selection were: between two and six years old, and diagnosed with ASD according to the ADOS. In addition to these criteria, we consulted with the therapist working with those children asking whether they thought the child would respond well to this type of scenario (e.g., sitting on a chair for a relatively long period of time). We also asked if there were any factors that may not make them suitable for our scenario (e.g., fear of robots).

The ages of our sample ranged between 2 years 9 months and 7 years 1 month ($\mu = 4.64$, $\sigma = 1.36$ years). Seven were male (63.6%) and four female (36.4%). Three children had low severity scores, six moderate and two severe. Three of the participants (27.3%) had interacted with a robot before (but not NAO) in the context of a separate study. All participants successfully completed the session, except for one who only completed the assessment phase.

2.3.2 Experimental procedure

One of the experimenter first obtained informed consent from the child's parent/primary caregiver for using the data collected for research purposes, and optionally to use media for public research communication. The experimenter then brought the child into the experiment room, along with their parent(s)/caregiver(s). Before initiating the session, the child was given ample time to explore the robot, and was encouraged to touch it and talk to it. During this initial time, the Wizard controlled the robot progressively using a library of pre-defined actions meant to attract the child in case of lack of interest, or to calm the child in case of fear or distress. After the child was seated and ready to interact with the robot, the semi-autonomous control of the robot was initiated. From there on, the experiment timeline outlined in Figure 2.3 started. The total session time ranged between 15 and 20 min approximately. Figure 2.4 shows some snapshots from different sessions.

The parents were instructed to minimally intervene, especially during the tasks, so as not to bias our results. During the tasks, the experimenter followed strict guidelines when intervention was needed. She only intervened to make sure the child was looking at the robot before the robot initiated the JATT task, and at the screen (or at least away from the robot) for the NAME task, both of which are important pre-conditions for the tasks we are studying.

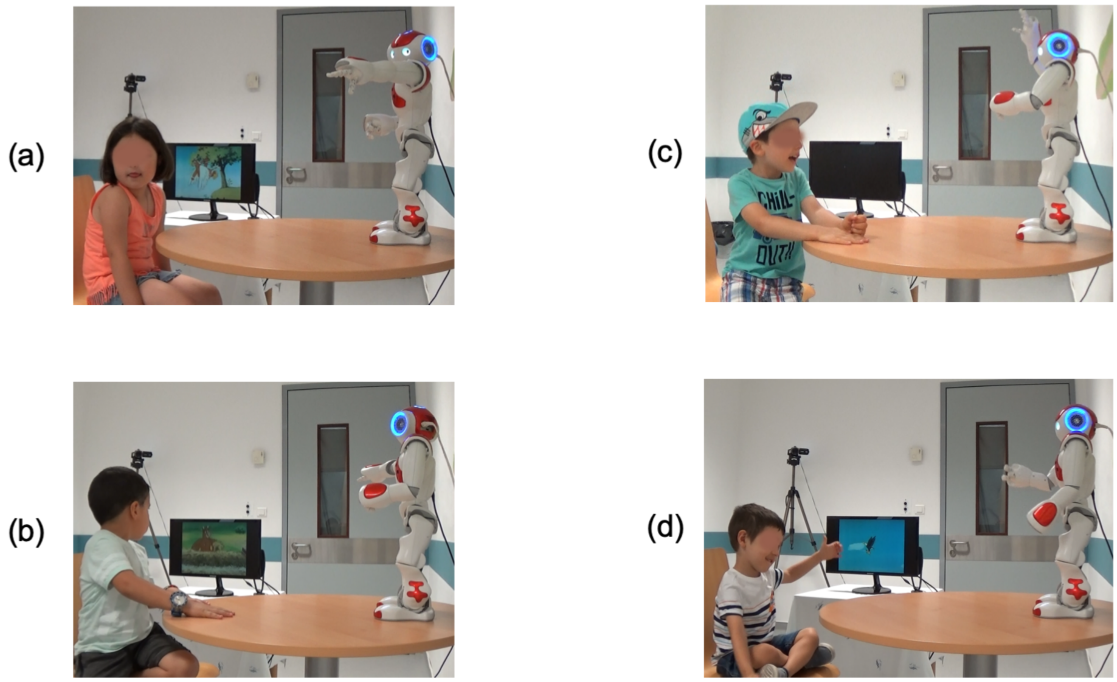

Figure 2.4 Snapshots from the experimental sessions. **(a),(b):** JATT successes for the right and left screen respectively (right screen not shown). **(c):** NAME success for action level 4. **(d):** Child imitating the robot's movement as instructed during storytelling for increased engagement. Images are shared under informed consent of parent/primary caregiver.

The role of the Wizard was played by a second experimenter during all the sessions. To ensure that there was no bias in the data he provided, we asked an autism therapist, who was agnostic to the aims of the study, to separately record her coding of children's responses for later comparison. Since she was not familiar enough with the Wizard interface, we decided that it was best for her not to operate the interface directly, as a low latency was crucial when triggering successes.

2.3.3 Counterbalancing

In the assessment phase, the choice of screen (left/right) in the JATT task alternated between consecutive tasks, and the choice of first screen was counterbalanced across participants. In the main interaction phase, the choice of screen was randomly selected while ensuring equal left/right proportions for each participant and not allowing more than two consecutive instances on the same screen, in order to avoid any practice effect. Also, the choice of the first mode in the alternating sequence (Therapy/Explore) was counterbalanced across participants.

2.4 Results

We extracted all relevant data from the robot logs, and analyzed them using a combination of SPSS, Matlab and Excel software. Our analysis mainly revolved around children's responses to the action sequences in the different robot modes.

2.4.1 Wizard coding method validation

We computed Cohen's Kappa interrater agreement between the Wizard's coding, which dictated the robot's behavior, and the coding of the autism expert present during the sessions. We compared the ordinal variables representing the index in the action sequence at which a success occurred. If no success occurred after exhaustion of the action sequence, we assigned a value of 5. If a success was triggered by the Wizard but not coded as a valid success by the expert, we assigned to the expert's coding a unique value (e.g., 0). Our analysis shows a high agreement between the two raters ($\kappa = 0.89, n = 248$). Based on observation, we believe that disagreements mainly occurred when the children exhibited multiple quick consecutive gaze shifts, which introduced ambiguity in coding.

2.4.2 Assessment results

Figure 2.5 reports the values of features RJA and RNA as assessed by the robot, according to Equation (2.1), and as assessed during an ADOS-based interaction with a human. Because the date at which the ADOS was administered differed significantly across children, we decided to re-assess the features of interest in an interaction with a human before the session. The assessment was done by one of the examiners who has experience with children with ASD and has a post-doctoral level training in psychological research. Some children had the ADOS administered the same week the study was run, so we did not reassess them and used the available ADOS features directly. Looking at the plots in Figure 2.5, the immediate observation is a difference in spread. A paired samples Friedman's two-way analysis of variance by ranks showed that the distributions between robot-assessed and human-assessed features are statistically significantly different ($\chi^2(1) = 11.00^{**}, p = 0.001$ for RJA, and $\chi^2(1) = 4.46^{*}, p = 0.035$ for RNA).

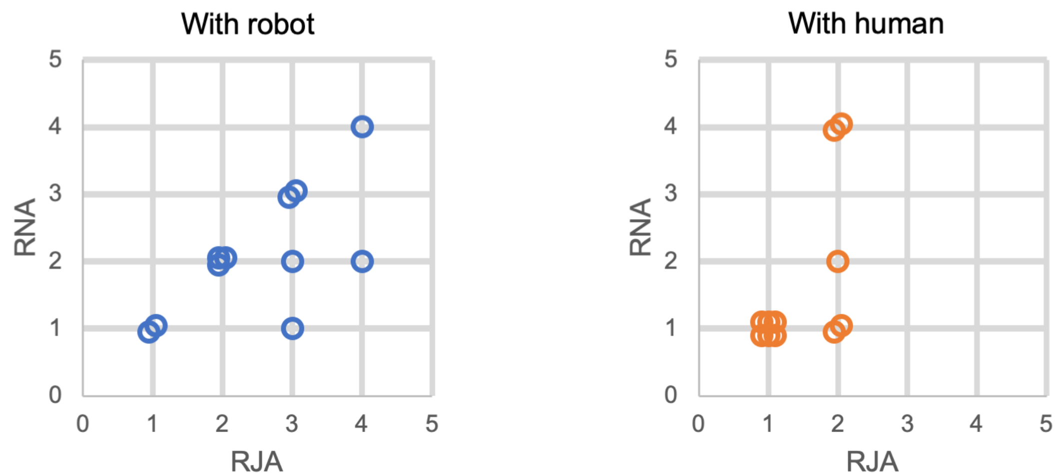

Figure 2.5 Distribution of children profiles during interaction with the robot and with a human in similar tasks. Overlapping points were slightly disturbed for better visibility. For comparison with ADOS feature values, reported values need to be reduced by one unit as ADOS feature values start at 0 by convention.

A Spearman correlation test showed a strong statistically significant correlation between the two robot-assessed features ($r_S(9) = 0.63^{*}, p = 0.037$). A similar correlation was found between the human-assessed features ($r_S(9) = 0.66^{*}, p = 0.026$). However, individual robot-assessed features did not statistically significantly correlate with the corresponding human-assessed features ($r_S(9) = 0.39, p = 0.236$ for RJA, and $r_S(9) = 0.30, p = 0.377$ for RNA). These results show that children's response to the human-administered tasks do not directly correlate to their response to a similar interaction with a robot, but that the cross-task response relationship is maintained. Moreover, the children overall needed significantly higher action levels with the robot than when interacting with a human ($\mu = 2.55$ versus $\mu = 1.45$ for RJA, and $\mu = 2.09$ versus $\mu = 1.64$ for RNA). In particular, while robot-assessed RJA values spanned all 4 possible values, the human-assessed values didn't exceed a value of 2. The restricted range for the human-assessed values was likely due to our selection criteria for children, whereby we favored children with enough attention span to follow a story.

In addition to RJA and RNA, an autism expert also coded the response to the four imitation prompts performed by the robot throughout the story. For each prompt, she coded the response into three ordinal categories (satisfactory (1), below average (2), and poor (3)). We then aggregated these responses into a single feature for each child according to Equation (2.1). A Spearman correlation test showed a strong and statistically significant correlation between the response to the robot's imitation prompts and the robot-assessed RJA ($r_S(8) = 0.73^*, p = 0.016$), as well as the robot-assessed RNA ($r_S(8) = 0.66^*, p = 0.037$).

In the ADOS, there is no feature specifically dedicated to response to imitation prompts (although there is a task that revolves around functional and symbolic imitation). Therefore, before the session the examiner who performed the reassessment of RJA and RNA also assessed imitation ability. She simply performed one of the prompts the robot would perform (namely, the 'Running' prompt) and asked the child to imitate her. She then coded the response in the same three categories as above. The response to robot-prompted imitation was not found to be statistically significant correlated with the response to human-prompted imitation ($r_S(8) = 0.40, p = 0.254$). Similarly to the results with RJA and RNA, children had statistically significantly poorer response on imitation with the robot than with the human, as shown by a Wilcoxon signed rank test ($Z < 0.001^*, p = 0.015$). Interestingly, the response to human-prompted imitation correlated with both the robot-assessed RJA ($r_S(8) = 0.73^*, p = 0.016$) and with the robot-assessed RNA ($r_S(8) = 0.66^*, p = 0.037$).

2.4.3 Comparison of modes

In analyzing the occurrence of successes across the different modes, we used four different metrics, reported in Table 2.2:

- *Within-4 success rate* — i.e., the percentage of task instances in which a success occurred within the exhaustion of a full action sequence.
- *Within-2 success rate* — i.e., the percentage of instances in which a success occurred within the first two trials of an action sequence.
- *Average number of trials* — i.e., the average number of actions the robot had to execute during a task instance.
- *Average successful action level* — i.e., the average level of all actions that caused a success.

Table 2.2 Comparison of success occurrences in the three modes.

Metric	**Assess** ($n = 44$)		**Therapy** ($n = 40$)		**Explore** ($n = 40$)	
	JATT/NAME	**Tot.**	JATT/NAME	**Tot.**	JATT/NAME	**Tot.**
Within-4 success (%)	97.5/100	**98.8**	100/72.5	**86.3**	100/87.5	**93.4**
Within-2 success (%)	70.0/77.5	**73.4**	80.0/62.5	**71.3**	97.5/65.0	**81.3**
Average #(trials)	2.40/2.00	**2.20**	1.55/2.45	**2.00**	1.25/1.98	**1.62**
Average successful level	2.43/1.95	**2.19**	2.58/2.18	**2.38**	2.70/2.64	**2.67**

Overall, the Assess mode required the lowest action level on average to achieve a success, but at the cost of the highest average number of trials. It also had the highest within-4 success rate, which can be explained by the fact that it was ensuring that the children were exposed to a maximum number of different actions. The Therapy mode needed lower action levels to achieve a success on average as compared to the Explore mode, but higher than the Assess mode. However it needed less trials than the Assess mode, and more than the Explore mode. On the other hand, it had the lowest success rate on both metrics, except for task JATT where it outperformed mode Assess. The Explore mode had the highest within-2 success rate and the lowest average number of trials, but at the cost of needing the highest action level on average to achieve a success. Note that our sample was not large enough to achieve statistical significance on most of the pairwise comparisons discussed above (as evaluated by both a repeated measures and a mixed-effects model). Therefore, we advise the reader to take them merely as suggestive comparative results to guide further investigation.

Since cross-task comparisons in the Assess mode were included in Section 2.4.2, we focus our cross-task analysis here on modes Therapy and Explore. In both these modes, the JATT task showed a high success rate (80% or above across the first two metrics). The NAME task, however, showed significantly lower success rate, according to a two-proportion Z test, both within four trials ($Z = 8.94^{**}, p < 0.01$), as well as within two trials ($Z = 7.69^{**}, p < 0.01$). Similarly, a Wilcoxon signed rank test showed that the median average number of trials per participant was significantly lower for the JATT task as compared to the NAME task for both Therapy mode ($Z = 36.00^{*}, p = 0.011$) and Explore mode ($Z = 28.00^{*}, p = 0.018$). This same test showed no statistically significant results when considering the average successful level per participant ($Z = 7.00, p = 0.236$ for mode Therapy and $Z = 24.00, p = 0.857$ for mode Explore).

Finally, it is important to stress that a comparison of the different modes in a single session does not provide any information about the long-term benefits of these modes. This comparison is merely informative of how sequencing affects the children's response along the static metrics we selected. The session was too short to expect any practice effects, and we did not find any evidence of such effects in our data. We reported all three modes in the table, although it is to be noted that a methodologically sound comparison can only be made between modes Therapy and Explore, since several scenario-related factors differ in the Assess mode.

2.4.4 Qualitative observations

We observed that the first contact with the robot was crucial in determining if the child will accept or refuse to interact with the robot. Several of our pilot attempts failed because of lack of care during this critical phase, which we corrected for the actual study. For future research, we highlight the need to very progressively integrate communication modalities to avoid negative reactions of the child in first-time encounters with a robot.

Children with ASD are a particularly challenging population to work with, as the slightest change in stimulus can cause a large difference in outcome. For example, the way the screens were flashing had a big influence on whether the children responded or not, so a lot of care had to be put in fine-tuning the scenario parameters during our pilots. These parameters included the screen behavior, the volume of the robot, the placement of the screens, the story length and content, the interval between consecutive task instances, the positioning of the imitation prompts, among other considerations that we iteratively refined.

Also, the variability across children also affected other aspects beyond the performance on the tasks. For example, while some children paid full attention to the story and were fascinated by the robot's behavior, others had moments of complete distraction. Such distraction moments included being fascinated by aspects of the interaction irrelevant to the story, like for instance fixating or touching a body part of the robot.

A final observation concerns a hypothesized relationship between engagement and practice effects. While there is no quantitative evidence of a consistent change in behavior throughout the session, we noticed that two effects seemed to balance each other differently across children. On the one hand, we noticed that engagement tends to plateau after a few minutes of interaction and then starts decreasing towards the last third of the session, which may worsen the performance of children on the tasks in that period. On the other hand, practice effects have the opposite effect level on task performance in that they increase it. Therefore, it is difficult to dissociate these two effects in our data, and future research could look at examining them separately to further inform robot adaptation within the session.

2.5 Discussion

The results of the study bring insight into the structure of children profiles, the effect of action sequencing on children's responses, and differences between tasks. We discuss each of these in turn, and end this section with additional thoughts.

2.5.1 Children profiles

Our comparison of profiles in the interaction with the robot versus with a human suggests that the information encoded in the human-administered ADOS cannot be used directly to inform an interaction with a robot. In addition to the lack of evidence for a correlation between the two, the children overall needed higher action levels with the robot than when interacting with a human. This result is in accordance with the existing literature on socially assistive robotics [152, 4].

This result can be explained by the lower degree of expressivity and naturalness of the robot as compared to a human, or by the lack of familiarity of the children with the robot's behavior. In particular when it comes to gaze, literature on general human response to robot gaze has also shown reduced reflexive gaze as compared to response to human gaze, which may have been a contributing factor in our JATT task [1]. It is also worth mentioning that the robot performed each action only once, while in the context of the ADOS actions are repeated several times to ensure a lack of response at a given level before moving to the next. In our scenario, we eliminated repetitions because we expected a very high number of trials to be harmful for engagement. However, we collected several measurements to reduce the effect of the incurred noise. These observations highlight the importance of having the robot perform its own assessment to be able to model the children's responses to its own actions accurately, and ensure the validity of personalized robot intervention such as in the Therapy mode.

On the other hand, our data showed significant correlations among robot-assessed features, including response to imitation prompts. These results may have implications on the development of more efficient methods for co-estimating those correlated variables, or for predicting cross-task performance from measurements on a single task.

2.5.2 Effect of sequencing

In this exploratory study, we analyzed the effect of sequencing on child response through controlling the robot mode. Based on our analysis, we observed that each of the modes comes with advantages and disadvantages.

First, the Assess mode favors using as low action levels as possible to cause a success. Based on our results, it seems to be well suited, beyond assessment, for cases where therapeutic goals need to be met with no concern for minimizing the number of trials. This applies when engagement and interaction flow are not priorities, for example in scenarios in which the tasks are repeated only a small number of times, or are sparsely distributed in time.

On the other extreme, the Explore mode seems to be a suitable mode if the only goal is to achieve early successes and to keep the child as engaged as possible. Its surprisingly high success rate, especially as compared to the Therapy mode, may be due to the high level of variability in action levels, which may cause children to respond more frequently. This effect could be explained by the existing literature on how statistically 'surprising' events lead to higher attention responses [81, 88]. Despite its high success rate, the Explore mode does not align with therapeutic principles of grading and just-right challenge characterized by consistency, scaffolding and gradual change in actions [134, 75]. As a result, it would not be suitable to be used for therapeutic purposes whose aim is to promote a positive change in response over time.

Between these two extremes, the Therapy mode aims to balance causing successes at low action levels and preferring a smaller number of trials. In the next chapter, we will be formalizing this tradeoff to generate optimal sequences that combine the advantages of these three modes into a single mathematical framework.

In short-term studies the novelty factor of the robot may have a strong effect on child response and may not reflect actual characteristics of the disorder, because as has been demonstrated in this work, the response to the robot greatly varied, regardless of ASD severity. In long-term studies, the novelty effect may disappear. In contrast, the engagement of the child may also decrease, so long-term studies should be looked at with care.

2.5.3 Cross-task differences

There are a few possible explanations as to why we observed cross-task differences across modes that were not consistent with the human-assessed children profiles. These include the objective difficulty of the prompts, the nature of the scenario, the relative interest of the children in the cartoons versus the robot, and the relative cartoon novelty as compared to the consistency of the robot's appearance. Identifying the exact causes or combinations of causes would need additional research.

Since the tasks we considered are quite generic and can be easily adapted to a range of different scenarios with different targets, we expect good generalizability of our general findings across similar scenarios. For example, any target object can be equipped with controllable lights and sound, to play the role of video and sound from our scenario, and we expect similar response patterns to hold across classes of similar tasks.

2.5.4 Robotic platform

The use of a NAO humanoid robot provided both advantages and disadvantages for the research goals of this chapter. On the positive side, the embodiment and size of the robot make it generally attractive to children with ASD, as demonstrated by several studies [4, 146, 70, 55, 153] and confirmed in ours. Moreover, its humanoid appearance and control of individual joints allowed for flexible gesturing options, while using speech and lights as additional expressive modalities. The multi-modal aspect of the communication was especially useful to allow the progressive integration of social cues throughout the action scales, and contributed to keeping the children engaged. On the negative side, the robot lacked actuated eyes, which limited the expressivity of the robot's gaze behavior. Furthermore, its speech was often monotonous or unclear despite our best efforts to make it engaging and articulate.

While we expect our general results to hold across different kinds of humanoid platforms, we hypothesize that the degree of anthropomorphism of the robot will have an effect on both the success rate of individual actions, and the discrepancy between the children's response to the human versus the robot. For example, we expect more responsiveness and less discrepancy between human-assessed and robot-assessed features for more anthropomorphic robots (e.g., robots with eyes, hair, or artificial skin). Additional research is needed to verify this hypothesis.

2.6 Related work

We present a brief overview of related work in robot personalization, attention-related tasks, and interactive storytelling with a focus on the autism domain.

2.6.1 Robotic personalization and adaptation in the ASD domain

While personalization and adaptation in Human-Robot Interaction (HRI) have been increasingly relevant topics across many different domains [127, 2], in the autism domain many existing approaches still heavily rely on tele-operation or content customization [113]. According to Esteban et al., higher levels of autonomy are needed to bootstrap the performance and flexibility of such systems [55]. The authors believe that supervised autonomy would be the ideal solution, leveraging the advantages that autonomy has to offer while including the therapist in the loop to ensure that the robot does not perform detrimental actions. Along these lines, in our semi-autonomous solution, the 'perception Wizard' plays the role of the human in the loop. The authors also developed a platform-independent architecture for personalization [34], but it has not yet been fully tested yet in the context of therapy. On the perception side, the personalization of algorithms for detecting child behaviors, for instance related to affect and engagement, has also been investigated [128]. These works highlight the importance of personalization in every component of a system developed for ASD intervention.

Real-time and long-term adaptation are other major aspects of socially assistive autonomy. Examples of adaptive system in the autism context include an affective robot adaptation method through multimodal measurements of affect to regulate a basketball-based task [43], and a model for graded robotic cueing in an imitation task [70]. Recent work by Clabaugh et al. (2019) demonstrated the effectiveness of their personalization and adaptation approach for in-home robot-assisted therapy [39]. In this chapter, the fact that the robot adjusts its action level when no success is observed can be seen as an example of basic adaptation. In the future, our approach could be extended to incorporate adaptation to account for the child's progress over time.

2.6.2 Attention-related robotic tasks

Several works have looked at robot-mediated solutions to train joint attention skills of children with ASD [4, 47, 80, 152]. In particular, Anzalone et al. (2014) investigated a spatio-temporal model of response to robots versus humans, showing generally lower response to robots [4]. Other work in this space has analyzed gaze patterns in different gaze orienting tasks [53] as well as non-verbal cognitive tasks [93].

The work that comes closest to the contributions of this chapter is that of Warren et al. (2015), who developed ADOS-inspired robotic tasks involving screens [152]. In our work, we used a similar setup, but while their focus was on studying the effect of the same action sequences throughout multiple sessions, our focus was to study and compare alternative action sequences. Also, while they focused on a single task (JATT), we additionally considered the NAME task, as well as imitation prompts. Furthermore, our study included a validation of response coding as well as a larger sample size. While some of the results in this chapter were in line with their findings, our main findings are novel as compared to the existing literature.

2.6.3 Robot-administered diagnosis

In addition to therapy, perhaps the most investigated use of robots in relation to autism is diagnosis. The idea is to take advantage of the objectivity and controllability of robots to reduce the variability and subjectivity of human-administered diagnostic and assessment.

The work in robot-administered diagnosis has considered several subproblems, such as developing quantitative metrics of social response [131], developing standardized tasks inspired by existing diagnostic tools [117], or researching algorithms for relevant robot perception [120, 90]. Petric et al. (2017) developed a framework for robot-administered diagnosis based on a Partially Observable Markov Decision Process (POMDP) formulation, to assess specific child features, using robot perception of multiple social and communication cues [118]. They developed four robotic tasks inspired by the ADOS, and tested them in a clinical setting [117].

In our work, we use a similar approach to robot control (mode Assess) to estimate a profile of the child for the purpose of subsequent personalization, and not for precise diagnosis. In Chapter 5, we provide an alternative role for robots in the diagnosis process by using the robot as a simulation platform for complementing the diagnostic training of therapists. We believe that the human aspect of diagnosis is an important one, as ultimately an evaluation of ASD should be measured with respect to a human rather than an artificial agent. In fact, our study results suggest that children do not respond in the same way to a robot as compared to a human in this context.

2.6.4 Robotic storytelling

Storytelling has often been used in HRI and technology-based scenarios, with both adults and children, for a wide array of educational goals [62, 92, 52, 147]. However, the interactive component of these scenarios remains limited, and has been the focus of recent investigation [114, 115]. In particular, Sun et al. (2017) introduced a collaborative storytelling scenario in which both the robot and the child contribute to create a story [143].

While most storytelling scenarios focus on the expressivity of the robot and the educational goals, in this work we introduced novel ways of introducing interactive engagement in robotic storytelling, through the use of screens that illustrate and support the story with engaging video snippets.

2.7 Summary

This chapter's main goal was to study how different sequencing of a provider robot's actions affects the response of a receiver (namely, a child with ASD). We studied this problem in the context of two robot-assisted attention-related therapy tasks inspired by the ADOS diagnostic tool. In a first step, we leveraged the structure of the ADOS tasks to build robotic actions on the NAO robot. We then integrated those actions into a control architecture that allows the robot to operate in three modes: Assess, Therapy, and Explore. These modes generate different sequences of the same robot actions, with different properties. To evaluate the effect of the different modes, we developed a semi-autonomous robotic scenario based on interactive storytelling, which integrates the tasks. Our data collected with 11 children with different ASD severities highlight the advantages and disadvantages of each mode depending on the interaction goals. The Assess mode favored a consistent and progressive evolution of action levels and had the highest therapeutic value, at the cost of a high number of trials. The Explore mode had the lowest number of trials but the least therapeutic value. The Therapy mode finds a tradeoff between meeting the conflicting goals of maximizing therapeutic value while minimizing number of trials.

The next chapter will aim at improving on the action sequences used in the Therapy mode by formalizing the said tradeoff. We will contribute optimal action sequences that consider therapeutic goals by preferring lower action levels (rationale behind mode Assess), while at the same time favoring smaller numbers of trials (rationale behind mode Therapy), and potentially incorporating the benefit of variability that characterizes the Explore mode. In that sense, these contributed sequences will fall somewhere between the three modes studied in this chapter.

Chapter 3

Optimal Action Sequences for Robot as Provider

Informed by the methods and results of Chapter 2, the goal of the present chapter is to formalize the problem of generating appropriate action sequences that are: (1) adaptive to action outcomes (success/failure), and (2) personalized according to the receiver profile. This general problem finds relevance in several other healthcare and education contexts, where providers need to choose from a list of actions with increasing levels of assistance organized along a scale. Table 3.1 shows two examples of such action scales in contexts other than ASD. The first one, taken from Linebaugh et al. (1997) [100], is a speech therapy task where cueing is used to assist patients suffering from aphasia, a disorder affecting speech production [73]. The second one, taken from Luckin et al. (2007) [102], is a scale of hints on a science problem. Inspired by the common structure of such assistive settings, we present a domain-agnostic framework to dictate the action selection of a robot acting as a provider in these settings.

3.1 Assumptions

In this chapter, we consider a probabilistic framework for modeling binary action outcomes (success/failure). Generally, actions of higher level in the scale are more likely to cause a success on the task. Hence, we can think of such scales as sets of actions ordered by *increasing success probabilities.* It is important to note that those success probabilities are different for each receiver, depending on their abilities. Our generic problem formulation considers scales with an arbitrary number of actions ordered by increasing arbitrary success probabilities.

In addition to success probabilities, we assume that provider actions have associated implicit *costs*. In a general assistive context, higher levels of assistance are typically associated with higher costs, such as energy, time, or other resources spent to provide the assistance. In therapy contexts, the concept of cost is more nuanced. Depending on the context and task, therapeutic costs may come from a number of factors, including explicitness, difficulty, or stimulus intensity. The more an action differs from what is considered desirable or natural, the higher its therapeutic cost because it is less likely to build the desired receiver skills over time [75]. In education contexts, costs could capture the amount of information revealed in a hint, or the difficulty level of a prompt. Although we practically expect costs to be increasing with increasing action levels, our problem formulation considers arbitrary positive action costs. Furthermore, as in Chapter 2, we assume that tasks have a finite horizon, corresponding to the maximum number of *trials*. This constraint can come from a number of factors, including time frame of a task or a session, engagement ability of the receiver, or resources available to the provider.

3.2 Chapter goal

This chapter contributes a context-independent method for generating optimal action sequences to be followed by a provider agent, i.e., action sequences with *minimum expected overall cost*. Every action in the sequence has a probability of failing, in which case the agent executes the next action, and a probability of succeeding, which is associated with a reward and no subsequent action execution. The agent does not know ahead of time when a success will occur but knows the action parameters (success probabilities and costs). Hence, it can reason under uncertainty to plan for action sequences that balance urgency to achieve a success and parsimony in the selection of actions according to their costs. Figure 3.1 illustrates this goal through a generic example.

In our solution to this problem, we devise *OASssistMe*, an algorithm that finds an optimal action sequence given a set of action success probabilities and costs, and rigorously analyze its mathematical properties [11]. We then present several extensions of the basic algorithm, relaxing the assumption that the action parameters are fixed [12]. In a second step, we instantiate our framework in the robot-assisted ASD therapy tasks

Table 3.1 Examples of action scales used in speech therapy and education, adapted from [100] and [102] respectively. Actions of higher level in the scale provide more assistance, hence have a higher success probability.

Domain	**Success (task goal)**	**Level**	**Action**
Speech therapy for aphasia	Patient retrieves word correctly	1	"What's this called?"
		2	Directions to state function of item
		3	Directions to demonstrate function
		4	Statement of function
		5	Statement and demonstration of function
		6	Sentence completion
		7	Sentence completion + silent articulation of first phoneme
		8	Sentence completion + vocalization of first phoneme
		9	Sentence completion + vocalization of first two phonemes
		10	"Say '——' "
Science education	Student enters units correctly	1	"You have entered the right numbers, but units are wrong. Look in the problem. Enter units now."
		2	"You have entered [...] wrong. Distance is in meters. Time is in seconds. Enter units now."
		3	"You have entered [...]. Distance is in meters and should be written as 200m. Time is in seconds and should be written as 25s. Enter that now.'

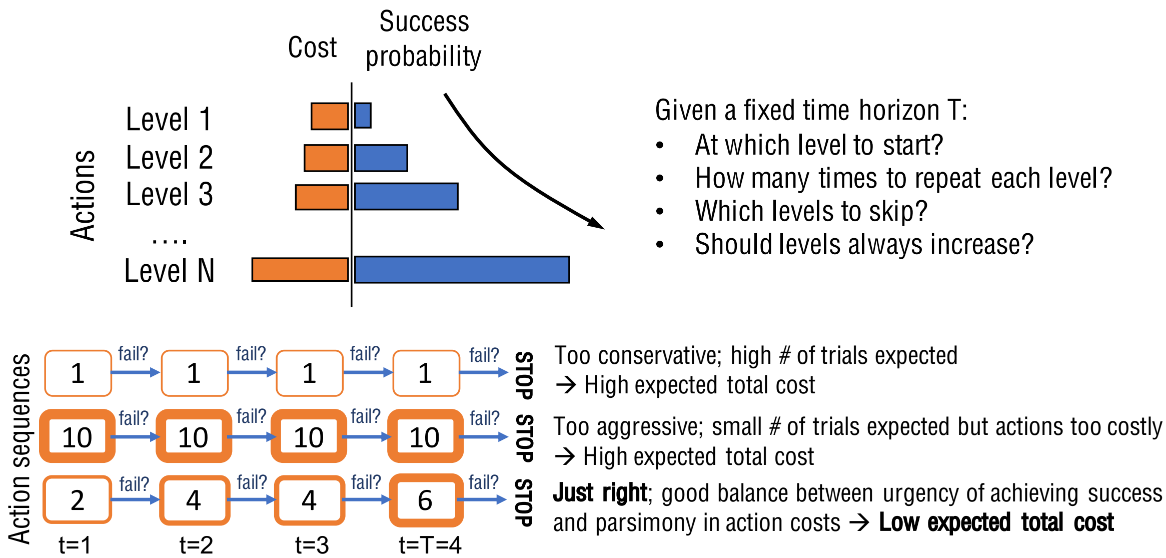

Figure 3.1 Generic example illustrating the concept of optimizing action sequences in relation to given action success probabilities and costs. Numbers in action sequences represent action levels. At every trial, in case of failure the agent continues the sequence, and in case of success gets a reward and aborts execution. Graphs and action sequences shown are only meant for illustrative purposes.

from Chapter 2. We first estimate action costs using expert data collected through an online survey with psychologists. We then estimate success probabilities for a given pre-assessed child profile, based on the real child-robot interaction data from Chapter 2. Our algorithm ultimately returns different optimal sequences for different child profiles, hence achieving "just-right" sequences for every receiver profile.

3.3 Mathematical framework

This section describes our contributed framework, which accounts for probabilistic outcomes of costly actions under a fixed time horizon. Within this framework, we present OAssistMe, an algorithm that generates optimal sequences, and analyze some of its properties.

3.3.1 Problem formulation

We frame the general problem informally defined in the beginning of this chapter as an optimization problem that takes into account both action costs and success probabilities.

Input

Assume we have a scale of actions $1, ..., N$ representing increasing levels of assistance. Further assume actions have fixed success probabilities $p(1) < ... < p(N) \in (0, 1)$ and costs $c(1) \neq ... \neq c(N) \in (0, \infty)$. Success probabilities of exactly 1 or 0 are not realistic and can lead to singularities in our analysis, which is why they are excluded. Additionally, one can argue that if two costs were equal the action with lower success probability should never be selected by an optimal agent, which makes that action irrelevant to the agent. The same argument applies for equal probabilities, in which case an optimal agent should always select the less costly action. As a result, we do not allow actions with equal probabilities and/or costs. Also note that while in application domains of interest we expect costs to be increasing, our problem formulation does not impose an order on the costs.

The outcome of every action a is assumed to be a Bernoulli random variable with success probability $p(a)$. We also assume there is a reward (negative cost) $R > 0$ associated with a success and no cost associated with a failure. Note that this last assumption does not compromise generality, since if failures are considered to be costly, the cost of a failure can be absorbed in the action costs and the value of R can be increased by the absolute value of that cost.

Setup

At each discrete trial $t \geq 1$, the agent selects an action a_t and observes the outcome. If a failure occurs, a new action is executed at the next trial. If a success occurs or the maximum number of trials (horizon) T is reached, the process stops. Trials are assumed to be independent, meaning the values of $c(a)$ and $p(a)$ are not influenced by previous actions in the sequence (later in Section 3.4, we relax this assumption).

Goal

The goal is to find an action sequence of length T that *minimizes the expected overall cost* of execution. The overall cost of an executed sequence is defined as the sum of costs of individual actions minus the reward R if a success occurs. Note that according to the setup above, the planned action sequence is only executed until a success occurs or the horizon T is reached, after either of which the agent stops. In the next subsection, we derive a closed form for the expected overall cost of a sequence, which corresponds to the objective function to be minimized.

Objective function

Let $\langle a_1, a_2, a_3, ..., a_T \rangle$ be an arbitrary action sequence. The probability P_t that a success occurs at trial t (upon which the agent stops) is given by:

$$P_t = p(a_t) \prod_{\tau=1}^{t-1} (1 - p(a_\tau)) \tag{3.1}$$

Note that for the same sequence we have the following recursive relation:

$$P_{t+1} = P_t \frac{p(a_{t+1})(1 - p(a_t))}{p(a_t)}, \quad P_1 = p(a_1) \tag{3.2}$$

Denoting C_t the cost of the sequence up to and including t:

$$C_t = \sum_{\tau=1}^{t} c(a_\tau) \tag{3.3}$$

We also have the following recursive relation:

$$C_{t+1} = C_t + c(a_{t+1}), \quad C_1 = c(a_1) \tag{3.4}$$

The expected overall cost of the actual sequence followed (aborted upon the occurrence of the first success) is hence given by:

$$O_T = \sum_{t=1}^{T} P_t(C_t - R) + (1 - \sum_{t=1}^{T} P_t)C_T \tag{3.5}$$

The first term represents all cases where a success occurs, while the second term represent the case where a success does not occur after all T trials (which is why it does not include R). An optimal action sequence $\langle a_1^*, a_2^*, ..., a_T^* \rangle$ is a sequence that minimizes the objective O_T.

3.3.2 Optimal sequence generation

We now present an algorithm to compute the solution to the optimization problem defined previously.

Single-trial case

For $T = 1$, the expected overall cost is $c(a) - p(a)R$, and the optimal action is $a^* = \arg\min_a \{c(a) - p(a)R\}$.

Multi-trial case

We can relate the objective O_T of sequence $\mathbf{\Pi}_T = \langle a_1, ..., a_T \rangle$ and the objective O_{T-1} of sequence $\mathbf{\Pi}_{T-1} = \langle a_2, ..., a_T \rangle$ (note the indices) as follows:

$$O_T = (1 - p(a_1))O_{T-1} + c(a_1) - p(a_1)R \tag{3.6}$$

Therefore, the optimal solution for horizon T can be obtained by first solving for the optimal solution for horizon $T - 1$ then appending at the beginning of the computed sequence the action a that minimizes the quantity $(1 - p(a))O_{T-1}^* + c(a) - p(a)R$, where O_{T-1}^* is the optimal objective function for horizon $T - 1$.

Hence, we have the following recursive relations,

$$O_T^* = \min_a \{(1 - p(a))O_{T-1}^* + c(a) - p(a)R\},\ O_1^* = \min_a \{c(a) - p(a)R\} \tag{3.7}$$

$$\begin{aligned} \mathbf{\Pi}_T^* &= \left\langle \arg\min_a \{(1 - p(a))O_{T-1}^* + c(a) - p(a)R\}, \mathbf{\Pi}_{T-1}^* \right\rangle, \\ \mathbf{\Pi}_1^* &= \left\langle \arg\min_a \{c(a) - p(a)R\} \right\rangle \end{aligned} \tag{3.8}$$

Based on Equations (3.7) and (3.8), we devise the OAssistMe algorithm (see Algorithm 1), based on dynamic programming. The resulting algorithm has linear time complexity in T and N ($\mathcal{O}(TN)$).

Algorithm 1 OAssistMe: Linear-time algorithm to find an optimal action sequence for horizon T.

procedure OASSISTME($\mathbf{p}$,$\mathbf{c}$,T,R) $\triangleright$ $\mathbf{p}$ and $\mathbf{c}$ are vectors containing $p(a)$'s and $c(a)$'s
    $\mathbf{O}_{part} \leftarrow \mathbf{c} - \mathbf{p}R$
    $\mathbf{O} \leftarrow \mathbf{O}_{part}$
    $O^* \leftarrow \min \mathbf{O}$
    $\mathbf{\Pi} \leftarrow \langle \arg\min \mathbf{O} \rangle$
    for $i \leftarrow 1, ..., T-1$ **do**
        $\mathbf{O} \leftarrow (1-\mathbf{p})O^* + \mathbf{O}_{part}$
        $O^* \leftarrow \min \mathbf{O}$
        $\mathbf{\Pi} \leftarrow \langle \arg\min \{\mathbf{O}\}, \mathbf{\Pi} \rangle$
    return $\mathbf{\Pi}$

For the sake of illustration, we present in Figure 3.2 a simple worked example with three generic assistive actions Low-assist – Medium-assist – High-assist, a horizon of 4, and sample action parameters.

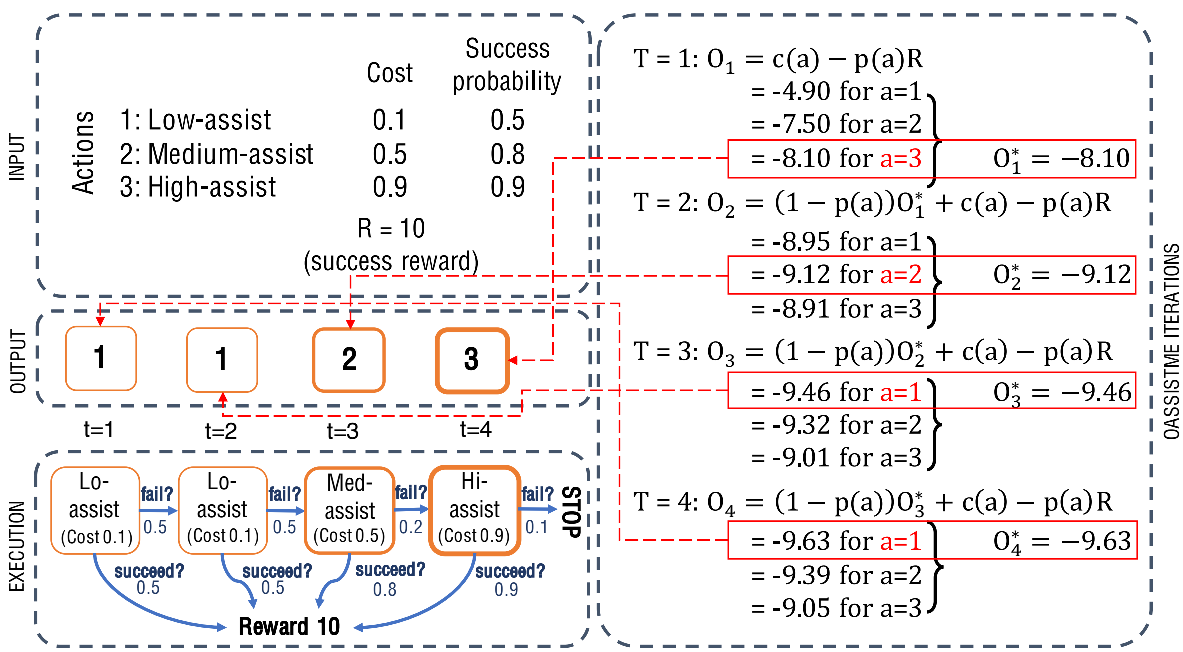

Figure 3.2 Worked example with $N = 3$, $T = 4$, and sample action parameters, showing: the computation of the optimal objective at every iteration, the resulting optimal action sequence, and the practical execution of the action sequence by the agent. Numbers below action outcomes represent probabilities.

3.3.3 Analysis of optimal solutions

We now present some theoretical results regarding optimal action sequences generated by the OAssistMe algorithm. We start by demonstrating a number of properties of optimal solutions, then briefly discuss a graphical representation of the problem and the role of the R parameter. We end by a brief interpretation of relevant results in relation to typical human provider strategies. In our analysis, we assume that ties are handled in a deterministic way, for example by always preferring lower actions.

Properties of optimal solutions

We present several properties of O_T^*, including monotonicity and convergence properties, and use those results to prove that *all optimal action sequences are monotonic* in t. Detailed proofs are included in Appendix B.

Our results are structured along the following three (mutually exclusive) cases:

(a) $O_1^* > 0$, or equivalently $R < \min_a c(a)/p(a)$.

(b) $O_1^* < 0$, or equivalently $R > \min_a c(a)/p(a)$.

(c) $O_1^* = 0$, or equivalently $R = \min_a c(a)/p(a)$.

Whenever the labels a., b., and c. are subsequently used, they refer to these three cases. The first result provides bounds on values for O_T^*:

Lemma 1. *For any T, we have one of:*

(a) $0 < O_T^* < \min_a c(a)/p(a) - R$.

(b) $0 > O_T^* > \min_a c(a)/p(a) - R$.

(c) $0 = O_T^* = \min_a c(a)/p(a) - R$.

Building on this result, we can show the following about O_T^* as a function of T:

Lemma 2. *O_T^* is monotonic in T. In particular, it is one of:*

(a) *strictly increasing, i.e., $O_{T+1}^* > O_T^*$ for all T.*

(b) *strictly decreasing, i.e., $O_{T+1}^* < O_T^*$ for all T.*

(c) *constant, i.e.,* $O^*_{T+1} = O^*_T$ *for all* T*.*

As a result, at every new iteration of the algorithm the computed value of O^* follows a consistent evolution, increasing, decreasing, or remaining constant, depending on the case, as the problem size is increased. Even though in practice horizons considered are relatively small, one might wonder, for the sake of better theoretical understanding, how such values behave at very large T.

Theorem 3. O^*_T ***converges*** *to* $\min_a c(a)/p(a) - R$ *as* T *goes to infinity.*

This result suggest that for T large enough, actions that are appended as T is further increased will stabilize to $\arg\min_a c(a)/p(a)$. In addition, for an infinite horizon, the optimal sequence becomes constant. In other words, an optimal agent should only select action $\arg\min_a c(a)/p(a)$ until a success occurs. Building on Lemmas 1 and 2, we now state our most important result regarding general properties of optimal solutions.

Theorem 4. *If* $\mathbf{\Pi}^*$ *is an optimal sequence, then it is monotonic in* t*. In particular,* $\mathbf{\Pi}^*$ *is one of:*

(a) ***nonincreasing****, i.e.,* $a^*_1 \geq a^*_2 \geq ... \geq a^*_T$*.*

(b) ***nondecreasing****, i.e.,* $a^*_1 \leq a^*_2 \leq ... \leq a^*_T$*.*

(c) ***constant****, i.e.,* $a^*_1 = a^*_2 = ... = a^*_T$*.*

Note that this result holds for any number N of actions such that $p(1) < ... < p(N)$, and for arbitrary costs $c(a) > 0$. For proofs of the four results above and how they build upon each other, we refer to Appendix B.

Graphical representation of O^*_T versus O^*_{T-1}

Some of the results presented above may be better understood with a graphical view on the problem. In the update function relating O_T to O_{T-1} (Equation (3.6)), every action a contributes a different linear relationship between the two quantities, with a different slope $(1 - p(a))$ and generally different y-intercept $(c(a) - p(a)R)$. As a result, according to Equation (3.7), the relationship between O^*_{T-1} and O^*_T is piecewise linear. Figure 3.3 shows a graphical representation of this relationship with sample costs and success probabilities falling in case (b). As can be noticed, changing the value of R

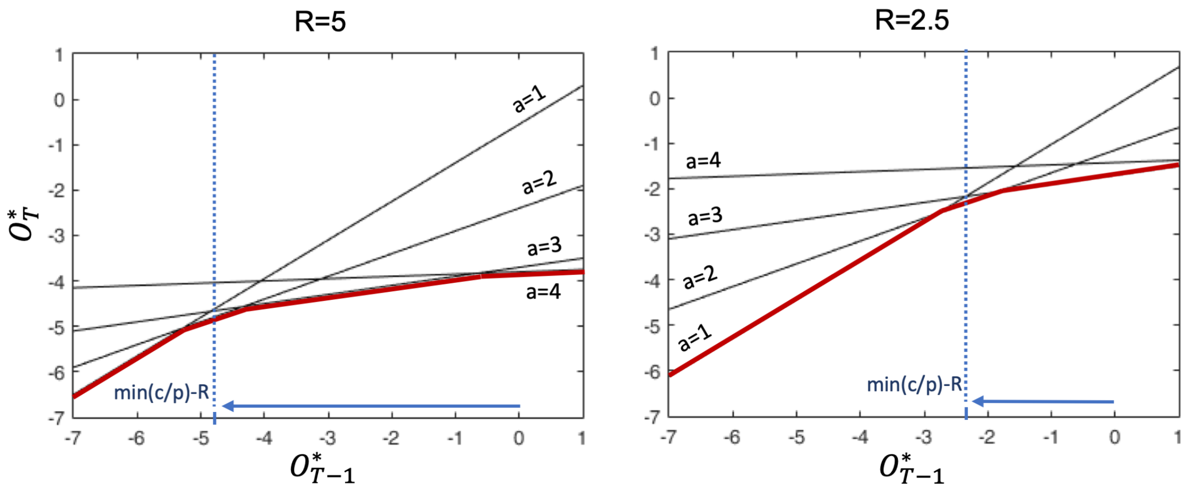

Figure 3.3 Sample graphical representation of O_T^* versus O_{T-1}^* as a piecewise linear function (red bold curve) for different R values falling in case (b) (greater than $\min_a c(a)/p(a)$). Arrows represent the direction of evolution of $O_{T-1}^* < 0$, and the dotted lines show the convergence point. The resulting optimal sequences for $T = 4$ are $\langle 2, 2, 3, 4\rangle$ ($R = 5$) and $\langle 2, 2, 3, 3\rangle$ ($R = 2.5$).

effectively translates the curve without changing its shape, nor the relative location of the convergence point.

Effect of R parameter

From the point of view of the generated optimal action sequences, the R parameter dictates how aggressively the agent tries to achieve a success. For all cases (a)–(c), increasing the value of R results in action sequences with equal or higher actions at every trial, and vice versa. This can be shown in a similar way to the proof of Theorem 4 (included in Appendix B), where instead of comparing the action selected at iteration T versus $T-1$, we compare actions selected at the same iteration but with different R values. As a result, we conclude that the R parameter effectively controls the *total probability of failure* $\prod_{t=1}^{T}(1 - p(a_t))$. Higher R values will result in lower or equal failure probability, and vice versa. In practice, one can set a threshold on the tolerance of failure and select the R parameter to meet that threshold.

Interpretation in relation to human provider strategies

In a typical provider-receiver interaction, a failure on the receiver side prompts the provider to repeat actions or increase assistance levels to gradually guide the receiver towards a success. This type of strategy is in accordance with the concept of grading in therapy [75], or scaffolding in education [66]. These results generally consist in adapting the assistance level according to the need and response of the receiver. Concretely, this means that the action sequences followed by human providers are typically nondecreasing.

This observation is consistent with our Theorem 4, case (b), which states that optimal action sequences generated by OAssistMe are not only monotonic but also nondecreasing. As a result, we conclude that in practice, *a value of R larger than* $\min_a c(a)/p(a)$ should be selected to incentivize increasing or maintaining assistance levels throughout the computed optimal sequences. In light of this result, in the rest of this chapter we will assume that $R > \min_a c(a)/p(a)$ for all practical uses of the OAssistMe algorithm.

3.4 Framework extensions

The framework presented in the previous section has relied on the assumption that action parameters (i.e., costs and success probabilities) are fixed. While costs are assumed to be intrinsic to the actions themselves and can be reasonably assumed to be fixed in a given domain, success probabilities may in practice possess some dependency on previous actions executed by the agent. As such, it is useful to consider the option that the success probability be a function of both the action executed at trial t and the history of actions *up to but not including* trial t, denoted by $\mathbf{h}_t = \langle a_1, ..., a_{t-1} \rangle$. We then denote the success probability function as $p(a, \mathbf{h}_t)$.

Generally, in the presence of dependence on history, the problem becomes a Markov Decision Process (MDP) where states contain an encoding of all possible histories $\mathbf{h}_t$ for $t = 1, ..., T$. Since this number, and as a result the number of model parameters, grow exponentially with the number of trials, it is desirable to identify what features of $\mathbf{h}_t$ may have an effect on the action parameters in practice. Inspired by our potential application domains, we consider three assumptions about how $\mathbf{h}_t$ can affect success probabilities. These assumptions are summarized in the three following cases:

1. $p(a, \mathbf{h}_t) = p(a, t)$, i.e., success probabilities are only affected by the number of previous trials, regardless of what actions were executed before the current trial. We call this case **trial-sensitive (TS)**.

2. $p(a, \mathbf{h}_t) = p(a, C_t)$, where C_t is the cost of sequence $\mathbf{h}_t$. We call this case **cost-sensitive (CS)**.

3. $p(a, \mathbf{h}_t) = p(a, n_t(a))$, where $n_t(a)$ represents the number of occurrences of action a in $\mathbf{h}_t$. We call this case **repetition-sensitive (RS)**.

We now motivate and discuss extensions of the framework to accommodate each of these cases, then analyze properties of optimal solutions as well as the time complexity of the extended algorithms. We end this section with a simulated example comparing each case to the basic case.

3.4.1 Trial-sensitive case (TS)

In a therapy and education context, it is somehow intuitive to consider a slight *positive* increase in success probabilities as a function of number of trials. For example, giving hints on an educational exercise may increase the likelihood of the student solving the problem correctly when the next hint is given. Similarly, in a therapy task that involves sensory integration [134], more trials translate into increasing overall sensory stimulation, which may make the receiver more likely to respond to individual stimuli.

For the sake of generality, the modifications to the original framework do not assume a specific relationship (e.g., positive, negative) between trial and success probabilities. With the success probability function $p(a, t)$ depending on both action and trial, the recursive relations described in Equations (3.7) and (3.8) become:

$$\begin{aligned} O^*_\tau &= \min_a \{(1 - p(a, T - \tau + 1))O^*_{\tau-1} + c(a) - p(a, T - \tau + 1)R\}, \\ O^*_1 &= \min_a \{c(a) - p(a, T)R\} \end{aligned} \tag{3.9}$$

$$\begin{aligned} a^*_{T-\tau+1} &= \arg\min_a \{(1 - p(a, T - \tau + 1))O^*_{\tau-1} + c(a) - p(a, T - \tau + 1)R\}, \\ a^*_T &= \arg\min_a \{c(a) - p(a, T)R\} \end{aligned} \tag{3.10}$$

where T is the specified horizon and τ represents the number of decisions left. The revised algorithm in this case, denoted by TS-OAssistMe, is summarized in Algorithm 2.

Algorithm 2 TS-OAssistMe: Trial-sensitive extension of OAssistMe where success probabilities are a function of trial.

```
procedure TS-OAssistMe(N, p_TS(.,.), c(.), T, R)
▷ p_TS is a function of action and trial and c is a function of action
    O* ← 0
    Π ← ⟨⟩
    for i ← T, ..., 1 do
        O ← ⟨(1 − p_TS(a,i))O* + c(a) − p_TS(a,i)R ,  a = 1, ..., N⟩
        O* ← min O
        Π ← ⟨arg min {O}, Π⟩
    return Π
```

3.4.2 Cost-sensitive case (CS)

In addition to the number of trials, the dependency on the history may be affected by previous action costs. Given that higher levels of assistance will typically be associated with higher costs, it may be the case that the success probability is *positively* affected by history cost. Back to our sample domains in the previous subsection, the success probabilities may for instance be sensitive to the total amount of information revealed by hints or cues (in an education or speech therapy context), or the total amount of stimulation provided (in a sensory integration therapy context).

As in the previous case, the modifications presented below do not assume that the relationship between cost of history and success probabilities has a specific form. With the success probability function $p(a, C_t)$ depending on both action and cost of history, we need to consider all relevant histories at every iteration of the algorithm. Because history cost is sensitive to the count of each action, but not the actual sequence order, we can represent histories as tuples $\langle n_t(1), n_t(2), ..., n_t(N)\rangle$, where $n_t(a)$ represents the number of occurrences of action a. The updated recursive relations now need to be applied to every distinguishable history at each iteration, as follows:

$$
\begin{aligned}
O^*_\tau(\mathbf{h}_{T-\tau+1}) = \min_a \{ & [1 - p(a, C_{T-\tau+1})]\, O^*_{\tau-1}(\mathbf{h}_{T-\tau+1} \cup a) \\
& + c(a) - p(a, C_{T-\tau+1})R\}, \\
O^*_1(\emptyset) = \min_a \{ & c(a) - p(a, 0)R\}
\end{aligned}
\tag{3.11}
$$

where the $\mathbf{h} \cup a$ operation adds action a to history $\mathbf{h}$ by incrementing $n_t(a)$.

Unlike in previous cases, the computation of the objective and the construction of the optimal action sequence are not performed in a synchronized way. Instead, the action sequence is obtained through backtracking after the computation of all O^* values is complete. For every computation of O^*, a corresponding action is stored. The final sequence builds from the first action in the sequence to the last action by appending the optimal actions successively, using the results from the backward pass. The revised algorithm in this case, denoted by CS-OAssistMe, is summarized in Algorithm 3.

Algorithm 3 CS-OAssistMe: Cost-sensitive extension of OAssistMe where success probabilities are a function of cost of history.

```
procedure CS-OAssistMe(N, p_CS(.,.), c(.), T, R)
  ▷ p_CS is a function of action and history cost and c is a function of action
    {H_1, ..., H_T} ← GenAllUnorderedHists(N, T)
  ▷ Generates all possible unordered histories H_i of size i − 1, represented as sets of
    tuples ⟨n_t(1), ..., n_t(N)⟩, where n_t(a) represents the number of occurrences of action a
    O*_{T+1} ← 0
    for i ← T, ..., 1 do
        for all h ∈ H_i do
            O_{i,h} ← ⟨(1 − p_CS(a, C_i)) O*_{i+1,h∪a} + c(a) − p_CS(a, C_i) R ,  a = 1, ..., N⟩
            O*_{i,h} ← min_a O
            Π̃_{i,h} ← arg min_a O
    Π ← BacktrackOptimalDecisions(Π̃)
    return Π
```

3.4.3 Repetition-sensitive case (RS)

There may be interesting effects linked to the repetition of the same action during an interaction with a receiver. For example, some research suggests that unpredictable ('surprising') sequences lead to higher attention responses [81], which may for example impact how patients respond to therapeutic tasks involving attention mechanisms [88]. This observation suggests that predictable sequences, such as ones that favor repeating previously executed actions over selecting new ones, may have a *negative* effect on success probabilities. In an education scenario, the same effect may be observed, where a hint is only helpful the first time it is shown. If a hint has been shown before and failed to cause a success, then it is reasonable to assume that subsequent trials of the same hint may have lower success probability.

As before, the modifications presented below do not assume that the relationship between number of repetitions and success probabilities has a specific form. The recursive relations are very similar to the CS case. The representation of history is identical since it needs to capture the count for each distinguishable action in the history, but is agnostic to the order. The updated equations are:

$$
\begin{aligned}
O^*_\tau(\mathbf{h}_{T-\tau+1}) = \min_a \{ & [1 - p(a, n_{T-\tau+1}(a))] \, O^*_{\tau-1}(\mathbf{h}_{T-\tau+1} \cup a) \\
& + c(a) - p(a, n_{T-\tau+1}(a))R \}, \\
O^*_1(\emptyset) = \min_a & \{c(a) - p(a, 0)R\}
\end{aligned}
\tag{3.12}
$$

The optimal action sequence construction is identical to the CS case. The revised algorithm in this case, denoted by RS-OAssistMe, is summarized in Algorithm 4.

Algorithm 4 RS-OAssistMe: Repetition-sensitive extension of OAssistMe where success probabilities are a function of the number of action repetitions. It is identical to Algorithm 3 except for the update equation in line 6.

1: **procedure** RS-OAssistMe(N,$p_{RS}(.,.)$,$c(.)$,T,R)
 ▷ p_{RS} is a function of action and repetitions of that action; c is a function of action
2: $\{\mathbf{H}_1, ..., \mathbf{H}_T\} \leftarrow$ GenAllUnorderedHists(N, T)
3: $\mathbf{O}^*_{T+1} \leftarrow \mathbf{0}$
4: **for** $i \leftarrow T, ..., 1$ **do**
5: **for** all $\mathbf{h} = \langle n_i(1), ..., n_i(N) \rangle \in \mathbf{H}_i$ **do**
6: $\mathbf{O}_i \leftarrow \Big\langle [1 - p_{RS}(a, n_i(a))] \, O^*_{i+1,\mathbf{h}\cup a} + c(a) - p_{RS}(a, n_i(a))R \ , \ a = 1, ..., N \Big\rangle$
7: $O^*_{i,\mathbf{h}} \leftarrow \min_a \mathbf{O}$
8: $\tilde{\mathbf{\Pi}}_{i,\mathbf{h}} \leftarrow \arg\min_a \mathbf{O}$
9: $\mathbf{\Pi} \leftarrow$ BacktrackOptimalDecisions($\tilde{\mathbf{\Pi}}$)
10: **return** $\mathbf{\Pi}$

3.4.4 General history-dependent case (G)

The structure of the CS and RS cases represent the most general way of incorporating dependence on history into the success probability function. These algorithms can easily be extended to the case where success probability is a function of an arbitrary number of features of the history, in addition to the action, of the form $p(a, \phi_1(\mathbf{h}_t), \phi_2(\mathbf{h}_t), \ldots)$. In this case, given the most efficient representation of history for the features considered, one can run Algorithm 3 or 4, with the appropriate representation of history and the appropriate success probability function with no additional modifications.

3.4.5 Analysis of OAssistMe extensions

We now discuss the applicability of the properties reported in Section 3.3.3 to the algorithm extensions discussed in this section. We also provide an analysis of the time complexity of the different algorithm versions for comparison.

Properties of optimal solutions

While increasing the horizon T in OAssistMe resulted in simply appending actions to the beginning of the optimal action sequence of size $T-1$, this is not necessarily the case in the extensions presented in this section. Depending on the strength of the dependency on history, the action sequences may change more or less considerably as larger horizons are considered. Furthermore, our experimentation with these algorithms shows that for small enough dependence on history features, the theoretical results from Section 3.3.3 generally hold. However, for larger values this is not necessarily the case. For example, we observed cases of non-monotonic values of O^*_T (Lemma 2 doesn't hold) when injecting high (monotonic) dependence on history features. Furthermore, cases of monotonic O^*_T do not necessarily translate into monotonic optimal action sequences (Theorem 4 doesn't hold). Referring back to Figure 3.3, one can understand the effect of these dependencies graphically. At every trial, the slope of the lines corresponding to each action are altered according to the corresponding parameters, which affects the update function of the objective function. When the shape of the piecewise linear function changes with respect to the current value of O^*_T, the theoretical properties of optimal solutions cease to hold.

These observations suggest complex interaction effects between the different components of these more elaborate models, and hence the analysis of their behavior is best achieved through simulation (see Section 3.4.6).

Complexity analysis

We now provide a brief discussion and visualization of the time complexity for the different algorithms presented.

- OAssistMe: As mentioned in Section 3.3.2, the time complexity of the algorithm is $\mathcal{O}(TN)$.

- TS-OAssistMe: The number of operations is identical to OAssistMe, but different probability values are used at every iteration. Therefore, the complexity of the algorithm is still $\mathcal{O}(TN)$.

- CS-OAssistMe: The total number of histories considered for the computation of O^* values is given by:

$$\sum_{\tau=1}^{T} \binom{\tau + N - 2}{\tau - 1} = \binom{N + T - 1}{T - 1} \leq \frac{(N + T - 1)^{T-1}}{(T - 1)!}$$

  The backtracking step is linear in T, and hence has negligible complexity. The total time complexity of the algorithm is therefore $\mathcal{O}(N\frac{(N+T-1)^{T-1}}{(T-1)!})$, assuming that values of O^* are accessible in $\mathcal{O}(1)$ time – e.g., through a dictionary.

- RS-OAssistMe: As in the cost-sensitive case, the time complexity of the algorithm is $\mathcal{O}(N\frac{(N+T-1)^{T-1}}{(T-1)!})$.

- G-OAssistMe: In the worst case, the histories are represented fully as ordered sequences of actions. The total number of histories in this case is given by:

$$\sum_{\tau=1}^{T} N^{\tau} = \frac{N(N^T - 1)}{N - 1}$$

  As a result, the complexity is $\mathcal{O}(\frac{N^2(N^T-1)}{N-1})$, which can be simplified to $\mathcal{O}(N^{T+1})$. Note that this result assumes negligible complexity for the computation of history feature(s).

Figure 3.4 shows a visualization of the different algorithm complexities for comparison.

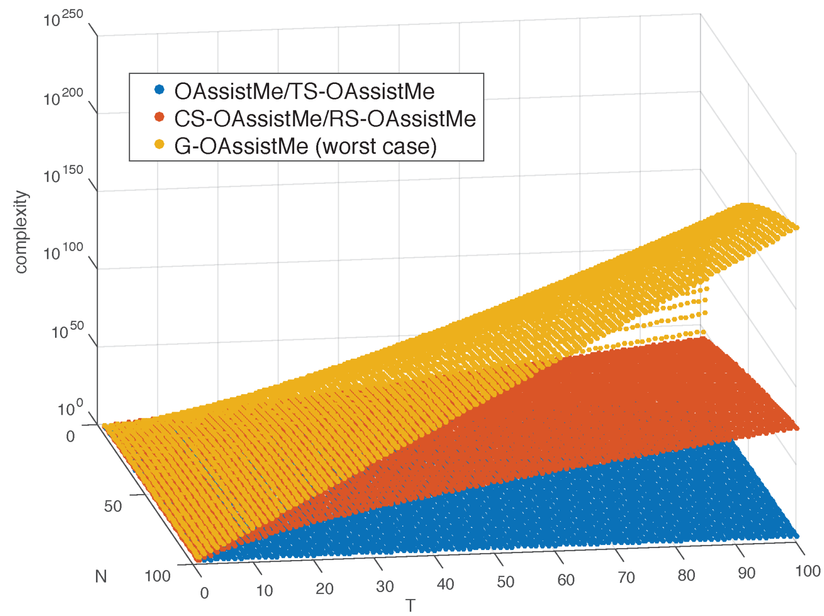

Figure 3.4 Time complexity curves for the different algorithms as a function of horizon (T) and number of actions (N). The G-OAssistMe case shows the worst case of possible histories to consider. Curves represent upper bounds on running times up to a constant factor. Plots are based on theoretical values and not experimental running times.

3.4.6 Simulated example

To evaluate the effect of our different assumptions about the relationship between history and success probabilities (TS/CS/RS), we consider a simulated example. Later in Section 3.5, we tie these frameworks to a therapy example.

We assume a logistic probability function of the form:

$$p(a, \phi(\mathbf{h}_t)) = [1 + e^{-(\beta_0+\beta_1 a+\beta_2\phi(\mathbf{h}_t))}]^{-1} \quad \text{for } a = 1, ..., N \tag{3.13}$$

where β_i's are weights and $\phi(\mathbf{h}_t)$ is the feature of history to consider (either t, C_t, or $n_t(a)$). In light of the motivations related to potential application domains included for each case (Sections 3.4.1, 3.4.2, and 3.4.3), we consider *positive* values for β_2 in the TS and CS cases, and *negative* values in case RS.

Furthermore, we assume a linear cost function of the form:

$$c(a) = \frac{c_{\max} - c_{\min}}{N - 1}(a - 1) + c_{\min}$$

Figure 3.5 shows sample action sequences generated by the different algorithms, showcasing the effect of increasing the weight of history dependency for each case.

On one hand, we can observe that TS-OAssistMe generally outputs more conservative action sequences as compared to OAssistMe. As the strength of the dependency (β_2) increases, the solutions become more conservative. This observation can be explained by the fact that as trials increase, actions become more effective at eliciting a success and hence generally lower actions are needed achieve a similar outcome.

On the other hand, solutions generated by the CS case are very similar to the ones generated by the TS case. The only difference occurs for a high value of β_2, where the algorithm is slightly more aggressive at trial 3 as compared to TS. This can be explained by the fact that the costs of the first two actions were relatively low to have actions deviate too significantly from the basic case. These observations suggest that the CS case captures similar aspects of history than the TS case, but with more resolution and hence can result in more intricate behavior depending on the action parameters.

Finally, the RS case generates outputs that seem to be less about how aggressive/conservative the algorithm is, but more about seeking 'novelty'. Even though differences are not obvious for low and medium values of β_2, for high β_2 the algorithm selects a new action at almost every trial. This observation can be explained by the fact that the algorithm is repetition-averse, as higher number of repetitions will decrease the agent's probability of achieving a success.

As part of our simulations, we also evaluated the effect of the R parameter on the optimal action sequences. Figure 3.6 illustrates the relationship between R and the total probability of failure $\prod_{t=1}^{T} (1 - p(a_t))$. In practice, one could use such a plot to inform an appropriate selection of the R parameter according to a tolerance threshold on failure probability.

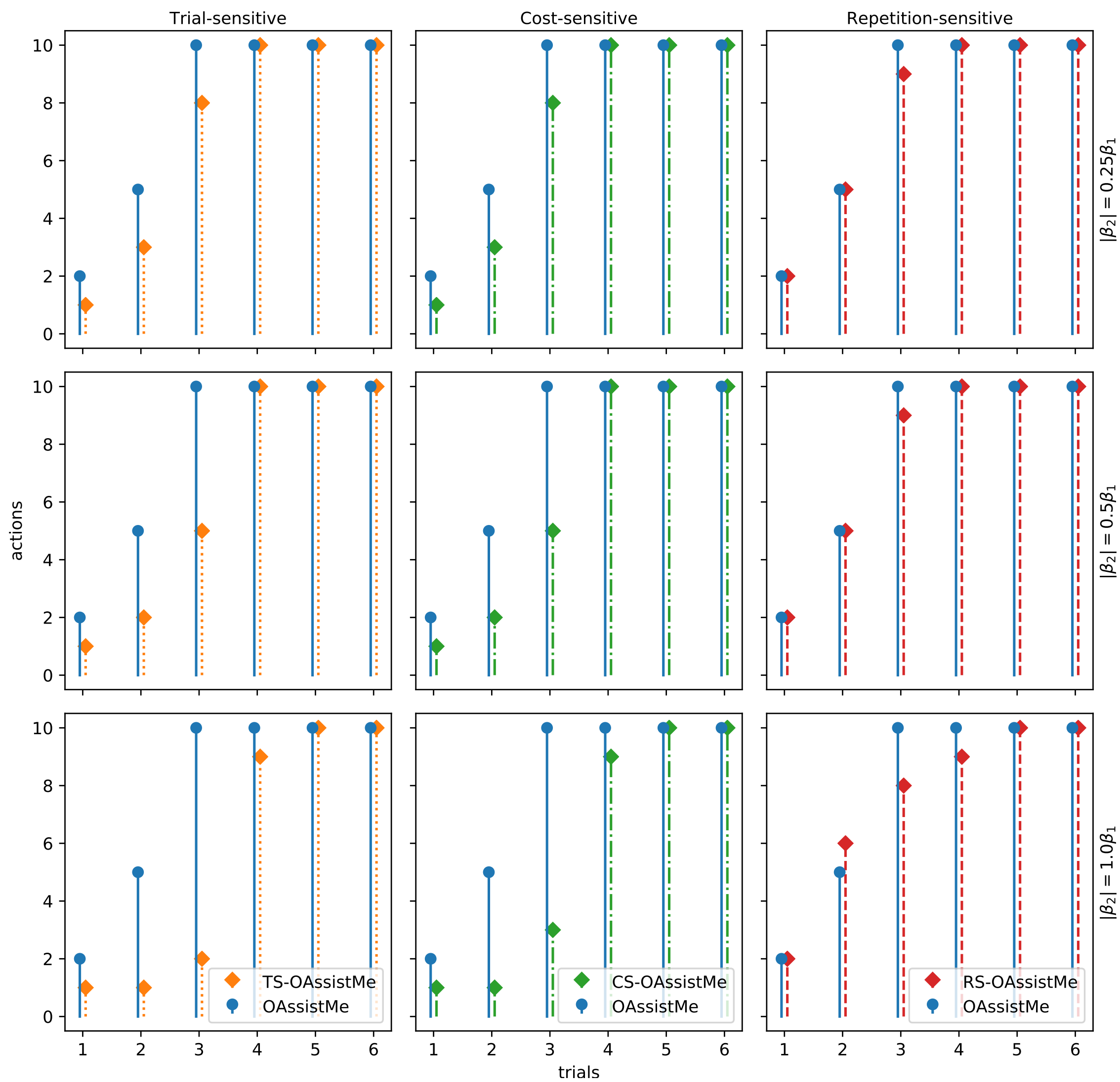

Figure 3.5 Illustrative comparison of optimal action sequences generated by the different versions of OAssistMe. *Parameters used*: $N = 10; T = 6; R = 10^4; c_{\max} = 1.7; c_{\min} = 0.8; \beta_0 = 0, \beta_1 = 0.25; |\beta_2| = 0.25\beta_1, 0.5\beta_1, \beta_1$ (positive for case TS and CS; negative for case RS).

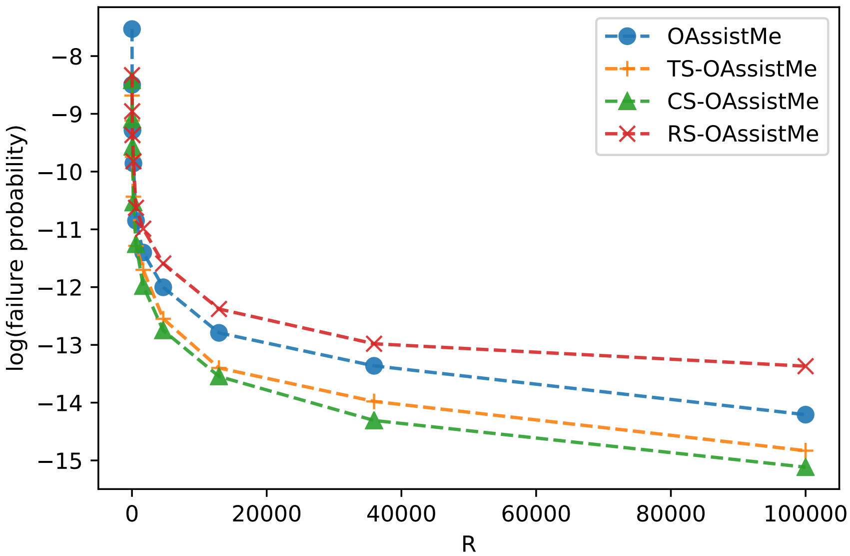

Figure 3.6 Effect of R on the total probability of failure for the different versions of the algorithm. Parameters used were identical to those in Figure 3.5.

3.5 Application to robot-assisted autism therapy

To illustrate the applicability of our theoretical framework, in this section we instantiate it in the robot-assisted therapy scenario from Chapter 2 (JATT and NAME). We provide a methodology for estimating action costs on the provider side (therapeutic robot), and success probabilities on the receiver side (child with ASD). We then use the estimated parameters to generate optimal action sequences for different child profiles corresponding to different levels of impairment, as assessed by the robot.

We now present our methodology for estimating: (1) the (therapeutic) costs of the robot actions, and (2) their success probabilities for different child profiles.

3.5.1 Cost estimation: expert survey

In a sensory integration context [134], such as autism therapy, it can be argued that the therapeutic cost comes mainly from how explicit a certain prompting or cueing action is. The more explicit the action – usually through the activation of more sensory channels, as is the case in our action scales – the further away it moves from natural everyday scenarios, which should be avoided. For these reasons, we use *level of explicitness* as our measure for action cost in this context, and we expect this measure to increase as the action level increases. Furthermore, we assume that the action costs, unlike the success probabilities, do not vary according to the receiver's abilities. They were hence measured with respect to what is expected for a virtual TD (i.e., non-ASD) child matching the age of our targeted population. The cost measured would then capture for each action its deviance from a natural interaction with a TD child.

To determine these action costs, we ran a video-based online survey where professionals in the fields of clinical, educational, and developmental psychology subjectively assessed the level of explicitness of our robot's actions shown as short video snippets. The responses for each robot action were gathered on a continuous scale (slider input) from 'Not explicit at all' (value of 0) to 'Completely explicit' (value of 100). Our sample consisted of 13 professionals from the areas of clinical (84.6%), educational (7.7%) and developmental (7.7%) psychology. Their ages ranged between 25 and 59 years ($\mu = 32.9$, $\sigma = 9.5$), and they were all female-gendered. Two participants completed only the first part of the survey, related to task JATT, and were included in the analysis. The participants were recruited through professional connections, and were not involved in the research project. Informed consent was obtained prior to showing the survey, whereby we explained that the aims of our research was to assess a robot's actions when interacting with a child, for the aim of informing robotic interactions in this context in the future. We gave them some background information on the task, stating that they were embedded in a storytelling task involving screens. We specifically asked them to answer the questions with respect to an imaginary TD child with the name 'Manuel' (which the robot used for the NAME task), aged between four and six years. 'Explicitness' was defined as how easy it would be for Manuel to understand the expected response to the robot's prompt. The survey was in European Portuguese and all participants were native speakers.

The collected data were analyzed using the SPSS software. The estimated costs and standard errors for each robot action are summarized in Table 3.2. Mauchly's test did not indicate any violation of sphericity neither for the JATT data ($\chi^2(5) = 9.63$, $p = 0.088$) nor for the NAME data ($\chi^2(5) = 6.44$, $p = 0.268$). A repeated measures ANOVA test showed no statistically significant differences between the mean costs for the JATT task ($F(3, 36) = 0.96$, $p = 0.423$), but showed statistically significant differences for the NAME task ($F(3, 30) = 4.82^{**}$, $p = 0.007$). A posthoc test with Bonferroni correction for multiple

Table 3.2 Mean estimated costs, along with standard errors, for actions in the two tasks based on experts' responses.

Action level	**JATT**		**NAME**	
	Cost	SE	Cost	SE
1	57.92	9.71	38.18	8.63
2	62.23	8.51	50.91	10.31
3	65.77	7.56	47.63	11.23
4	74.85	7.46	72.73	9.90

comparisons yielded statistical significance only between levels 1 and 4 ($p = 0.044$) for the NAME task. To measure inter-rater reliability, we calculated the intra-class correlation coefficient (ICC) based on a mean rating, one-way random effects model. We included both tasks in our analysis, and excluded two of the 13 participants who had a few missing items. The ICC estimate was 0.37 with a 95% confidence interval from -0.55 to 0.85 ($F(7, 80) = 1.58$, $p = 0.150$). This relatively low reliability value may be attributed to the different backgrounds of the raters, and their varying experience working with tasks similar to the ones considered.

The cost function follows an increasing trend along the scale for both tasks, as expected, with the exception of action 3 in the NAME task, which records slightly lower cost than action 2. The only difference between the two actions is the presence of lights, which may have been hard to notice on the video version. Given that the standard errors are high, we attribute this result to noise. However, it does not violate the assumptions of our framework, since the latter is valid for arbitrary positive cost functions.

3.5.2 Success probability estimation: child-robot interaction data analysis

In this section, we use the interaction data collected in Chapter 2 to determine a set of success probabilities that accounts for different *child profiles* within the two tasks JATT and NAME. As described in Chapter 2, the child profile is a categorization of the child's response to robot prompts into one of four discrete levels 1–4, according to the average action level at which the child successfully responds (Equation 2.1). We refer to these four levels in this chapter as: High response (1), Medium response (2), Low response (3), and Minimal response (4). According to the ADOS, higher values are typically associated with higher levels of impairment in attention mechanisms. The profiles were estimated by the robot during the assessment phase of the study described in Chapter 2. In the probability estimation analysis that follows, we only consider the sequence data obtained from Mode Explore (random), as it avoids biasing our estimation with any order effects.

Success probability model

Similar to our simulated example from Section 3.4.6, we use a logistic model of success probability according to the following equation:

$$p = (1 + e^{-\boldsymbol{\beta} \cdot \boldsymbol{\phi}})^{-1} \tag{3.14}$$

where vector $\boldsymbol{\phi}$ contains the predictor variables, in this order: constant term, child profile, action, and possibly a feature of history (trial, cost or repetitions), while vector $\boldsymbol{\beta}$ contains the feature weights. The inclusion of the child profile as a predictor variable enables us to accommodate for a range of different children. In order to determine which version of OAssistMe is best suited for this domain, we consider trial, cost of history, and number of repetitions as potential additional predictors, and fit the model to the data using multiple logistic regression. Prior to running the regression, the data were checked for potential learning effects across task instances for the same child, but no significant learning effect was found.

Our regression results, summarized in Table 3.3, show that while action level and child profile are statistically significant predictors, incorporating additional predictors does not significantly improve the model. We conclude that there is no evidence in this particular domain of an effect of history on success probability, at least given the amount of data at hand. Therefore, the basic version of OAssistMe is best suited given this data. Figure 3.7 shows visualizations of the regression results with action level and severity as the two predictors. Each data point (blue dots in the two upper plots) represents the average estimated success probability for a given child and action level.

We can see that the NAME task was overall identified to be more difficult since it had lower success probabilities, as well as lower costs (see Table 3.2). As a result, the total number of observations was higher in the NAME task ($n = 79$) as compared to the JATT

Table 3.3 Multiple logistic regression results on the child-robot interaction data, including predictor weights ($\boldsymbol{\beta}$), p-values and residual deviance of the fitted model (d). The first element in the weight vector corresponds to the constant term, while subsequent elements correspond to the included predictors in the order mentioned. Single/double stars mean significance to the 0.05/0.01 level.

Predictors	**JATT**	**NAME**
Profile + Action	$\boldsymbol{\beta} = \begin{bmatrix} 2.20 \\ -0.98^{*} \\ 0.79^{*} \end{bmatrix}, p = \begin{bmatrix} 0.138 \\ 0.043 \\ 0.035 \end{bmatrix}$ $d = 40.98$	$\boldsymbol{\beta} = \begin{bmatrix} 1.88 \\ -1.74^{**} \\ 0.65^{*} \end{bmatrix}, p = \begin{bmatrix} 0.079 \\ < 10^{-3} \\ 0.013 \end{bmatrix}$ $d = 83.04$
Profile + Action + Trial	$\boldsymbol{\beta} = \begin{bmatrix} 2.10 \\ -0.99^{*} \\ 0.79^{*} \\ 0.10 \end{bmatrix}, p = \begin{bmatrix} 0.196 \\ 0.044 \\ 0.035 \\ 0.883 \end{bmatrix}$ $d = 40.96$	$\boldsymbol{\beta} = \begin{bmatrix} 1.93 \\ -1.66^{**} \\ 0.69^{*} \\ -0.18 \end{bmatrix}, p = \begin{bmatrix} 0.073 \\ 0.001 \\ 0.011 \\ 0.546 \end{bmatrix}$ $d = 82.67$
Profile + Action + Cost	$\boldsymbol{\beta} = \begin{bmatrix} 2.19 \\ -1.04^{*} \\ 0.81^{*} \\ 0.25 \end{bmatrix}, p = \begin{bmatrix} 0.138 \\ 0.036 \\ 0.035 \\ 0.537 \end{bmatrix}$ $d = 40.56$	$\boldsymbol{\beta} = \begin{bmatrix} 1.83 \\ -1.71^{**} \\ 0.66^{*} \\ -0.03 \end{bmatrix}, p = \begin{bmatrix} 0.093 \\ 0.001 \\ 0.013 \\ 0.816 \end{bmatrix}$ $d = 82.98$
Profile + Action + Repetitions	$\boldsymbol{\beta} = \begin{bmatrix} 2.20 \\ -1.03^{*} \\ 0.835^{*} \\ 0.567 \end{bmatrix}, p = \begin{bmatrix} 0.138 \\ 0.043 \\ 0.037 \\ 0.734 \end{bmatrix}$ $d = 40.86$	$\boldsymbol{\beta} = \begin{bmatrix} 1.81 \\ -1.61^{**} \\ 0.61^{*} \\ -0.76 \end{bmatrix}, p = \begin{bmatrix} 0.091 \\ 0.001 \\ 0.019 \\ 0.358 \end{bmatrix}$ $d = 82.01$

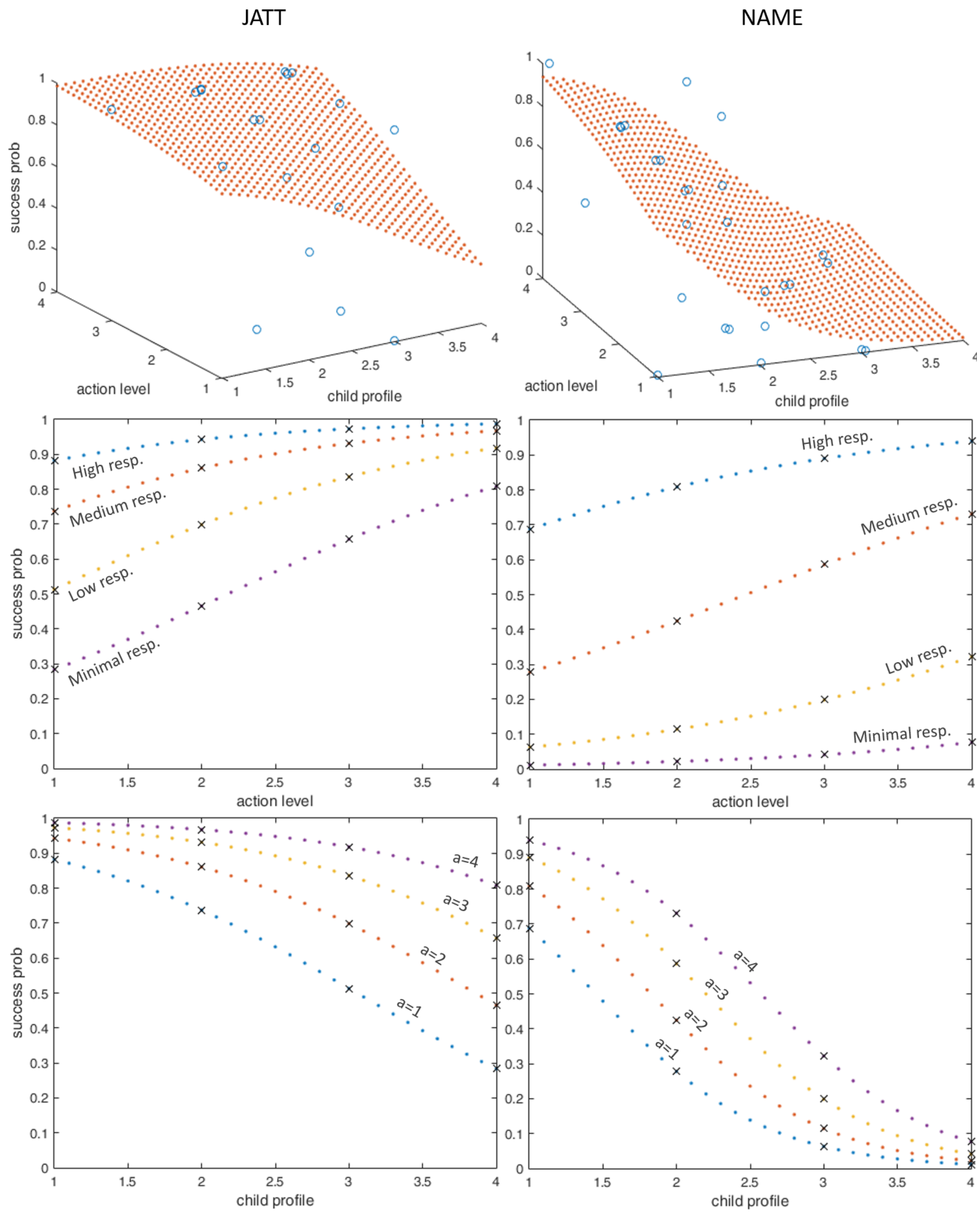

Figure 3.7 Success probability results showing data points and fitted surfaces (top), cross-sections relating $p(a)$ to a (middle) and cross-sections relating $p(a)$ to child profile (bottom). Overlapping data points are perturbed for better visualization. Continuous surfaces and curves are shown for illustration purposes. These results were obtained with MATLAB's GLMFIT function with a logit link function and binomial distribution of response variable.

Table 3.4 Minimum R values for acceptable algorithm performance for different child profiles on each task.

Task	Child profile	$R_{\min} = \min_a c(a)/p(a)$
JATT	High response (1)	65.68
	Medium response (2)	70.62
	Low response (3)	78.69
	Minimal response (4)	92.64
NAME	High resp. (1)	53.21
	Medium resp. (2)	80.81
	Low resp. (3)	225.87
	Minimal resp. (4)	948.24

task ($n = 50$) because successes occurred less frequently and actual sequences executed by the robot were longer, resulting in a smoother spread in the response variable.

Optimal action sequence results

The results presented above allow us to generate personalized optimal sequences according to the profile of each child. The only remaining parameter to determine is R, which can be tuned. According to the results presented in Section 3.3.3, we should choose $R > \min_a c(a)/p(a)$. Table 3.4 reports the minimum values of R for the different child profiles. Similar to action costs, R is a parameter intrinsic to the task, so we assume that it does not depend on the child profile. Therefore we should select a value of R greater or equal to the values reported in the table. For the purposes of this work, we will set R to 950 for both tasks (rounding up the largest value in the table $R_{\min} = 948.24$). In practice, one may want to consider different values of R for different tasks, depending on the relative importance of the skills that the task targets. As mentioned in Section 3.4.6, the selection of R in practice can also be informed by looking at how it affects the total probability of failure, as shown in Figure 3.8.

We ran the OAssistMe algorithm with the estimated action parameters, for both JATT and NAME tasks, and report the resulting optimal sequences in Figure 3.9. As expected, as the child profile increases, the computed sequences have generally higher or equal action levels. As mentioned previously, the NAME task was determined to be more

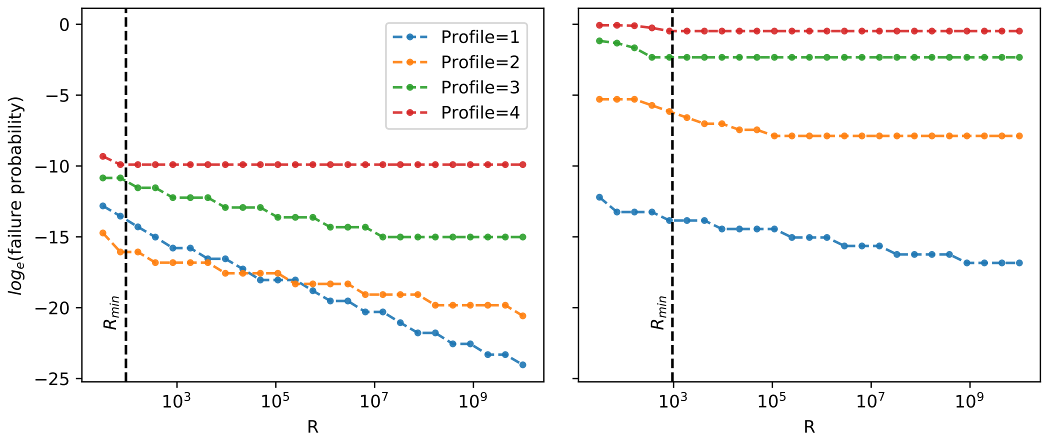

Figure 3.8 Effect of R parameter on total probability of failure for task JATT (left) and NAME (right), to inform R value selection. For every point, we first find an optimal action sequence for the corresponding action parameters, and then compute the reported total probability of failure value by multiplying failure probabilities for individual actions.

challenging than JATT, which may explain the overall higher action levels for all child profiles in NAME as compared to JATT.

In comparison with the action sequences from mode Therapy (Chapter 2), we observe that OAssistMe chooses equal or higher action levels across all cases. This result highlights the fact that Mode Therapy was more conservative than OAssistMe in its action selection. Additionally, to get a sense of the sensitivity of the algorithm to estimation error, we tested the inclusion of response data from modes Therapy and Assess when performing logistic regression. For all combinations 'Explore+Therapy', 'Explore+Assess', and 'Explore+Assess+Therapy', the action sequences computed by OAssistMe were identical, indicating some robustness to estimation errors. This property was not observed however in the extended versions of the algorithm, most probably due to the much larger data requirements needed to properly estimate parameters with three or more predictors. For purely informational purposes, we have included in Appendix B a complete report of the output of TS-OAssistMe, CS-OAssistMe, RS-OAssistMe, and a version of the algorithm that considers all predictors.

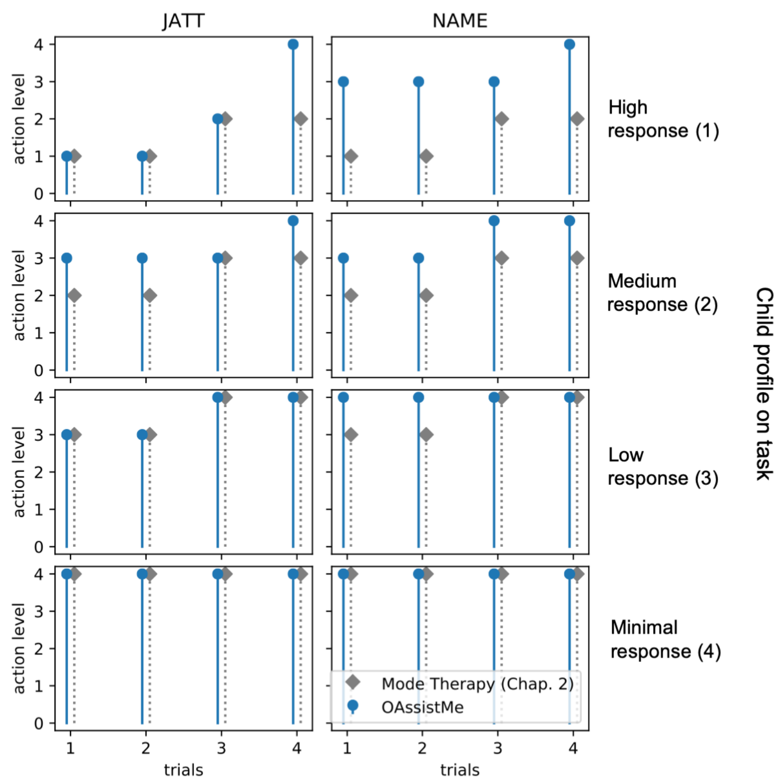

Figure 3.9 Sequences generated by the OAssistMe algorithm for $T = 4$ and $R = 950$ (sequences from Mode Therapy in the previous chapter included for comparison). For additional comparison, running the algorithm with uniformly spaced action costs (12.5, 37.5, 62.5, 87.5) and success probabilities (0.125, 0.375, 0.625, 0.875) yields the sequence $\langle 4, 4, 4, 4 \rangle$.

3.6 Discussion

We start by highlighting the major findings discussed in previous sections, proceed to state some of the limitations of our approach, and end by discussing the applicability of the algorithmic contributions of this chapter to other settings.

3.6.1 Major findings

The main theoretical results of practical relevance are:

- For high enough R, optimal sequences generated by the OAssistMe algorithm are *nondecreasing*, i.e., the agent should only maintain or increase the action level at the next trial if a failure occurs. This result aligns with typical strategies followed by providers.

- The R parameter affects the *total probability of failure*, hence having a threshold on this probability can inform an appropriate choice on R in practice.

The main observations based on our simulations of the framework extensions are:

- The assumption that success probabilities increase as a function of trial (case TS) seems to generally make optimal action sequences more *conservative*.
- The assumption that success probabilities increase as a function of cost (case CS) of history has a similar effect as the TS case, but allows for *more fine-grained algorithm behavior* by incorporating a cumulative effect of previously executed action levels.
- The assumption that success probabilities decrease as a function of repetitions seems to generally make optimal action sequences *repetition-averse*, hence more *diverse* (higher number of distinct actions).

These general observations are based on our experimentation with realistic parameters. The claims being sensitive to parameter selection, they should only be considered as suggestive results.

Finally, instantiating our framework in a robot-assisted autism therapy scenario leads us to the following observations:

- The action sequences generated by OAssistMe with the estimated action parameters show how our framework can achieve *personalization* to accommodate a range of receiver profiles.
- The two tasks considered show *different levels of difficulty* as reflected by differences in both estimated success probabilities and estimated action costs.
- While child profile and action level were confirmed to be significant predictors of success probability, our data did not show evidence of potential additional history effects such as trial, cost or repetition sensitivity. As a result, the *basic version of OAssistMe* was best suited for generating optimal sequences given the data at hand. However, the results from Chapter 2 (especially in Mode Explore) seemed to

suggest that there such dependence exists (specially repetition-sensitivity). The lack of evidence in our logistic model to support that explanation may simply be due to the small number of samples, as the data requirements increase significantly as more predictors are added to such a model. We hypothesize that these effects are more subtle that the ones due to severity or action level, and could be better investigated with larger sized studies.

3.6.2 Limitations

Despite our efforts to follow appropriate methodology in evaluating our framework, our approach does not come without some limitations. The results presented in the evaluation section of this chapter are preliminary and require further testing before they can be used in practice.

First, the action costs were assumed to be identical for all individuals. This assumption may be valid in cases where the cost is purely intrinsic to the action itself – e.g., execution time, financial cost, energy spent. However, it becomes fuzzier when the measure of cost possesses some level of subjectivity. In our domain, the variance in the experts' estimated cost values was high, which highlights this subjectivity. In order to reduce rater subjectivity, in our approach we measured the costs in relation to a virtual reference profile. On the other hand, assuming a profile-dependent cost on top of profile-dependent success probabilities could unnecessarily complicate our model, and may not even be desirable. It is important to note however that our algorithm was able to generate different action sequences for different child profiles, suggesting that assuming constant costs did not compromise flexibility. Moreover, our framework allows for the R parameter to be adjusted on an individual basis if the importance of succeeding on a given task or the tolerance to aggressive strategies differs across receivers with different needs.

Second, the survey data collected showed high variance and low reliability, which calls for more reliable methods to estimate action costs in these types of domains. One possibility would be to ensure that the participants have enough understanding and experience with the tasks and scenarios described, and to have baseline questions to test that their understanding of the measure aligns with the researcher's intended meaning. The validity of a general questionnaire approach to cost estimation could potentially be tested against an inverse reinforcement learning approach where costs are estimated directly from expert demonstrations.

Third, the analysis of the interaction data made an assumption of stationarity across instances of the same task. Even though no main learning effects were found in the data, some subjects did exhibit inconsistent behaviors across instances, such as disengagement, distraction, etc., which may have affected our results. In principle, if one is given a model of evolution of receiver response across several instances, then one can update the action parameters according to that model and run the same algorithms simply with a different input. However, since this chapter looked at a small number of task instances, it is not concerned with coming up with such models.

Fourth, even though the study presented in Section 3.5.2 is the first to collect this type of data with children with ASD under careful methodological considerations to reduce bias, it suffers from a low number of samples, as in most probabilistic frameworks. Specifically, because the number of data points for each action level and individual was low, the resulting response variable in our regression model showed a high spread. Higher number of samples per participant may result in better fit of our regression model but may also induce bias in our data due to potential positive or negative learning effects. Furthermore, the samples used for our logistic regression were not from fully independent data, and hence regression results may not be used for principled hypothesis testing purposes. The purpose of the regression in this chapter was merely for prediction and suggestion of appropriate algorithm selection. All of the questions discussed above should be kept in mind when designing similar data collection scenarios in the future.

3.6.3 Applicability to other settings

The algorithms presented in this chapter assumed an assistive interaction with binary action outcomes (success/failure). In some applications, the measure of success may however be non-binary. For example, one may consider the speed at which the child responds as a continuous measure of quality of response. In this case, the probability of success would have to be substituted by a probability distribution over continuous outcomes (or a probability mass distribution if the multiple outcomes are discrete). An outcome reward (negative cost) at every trial could also be included as a function of the outcome, and embedded into the objective function. The stopping condition for the robot, on the other hand, would still be defined in a binary way, i.e., either the child responds within the timeout range or they don't.

Another possibility beyond binary outcomes would be transitions to different states. In our extensions, we have explored the idea of states only in relation to features in history. In practice, a robot action could cause the receiver to transition to a specific state where its response changes drastically. For example, there is a probability that the child gets completely distracted and stops responding to the robot due to this distraction, and not due to an underlying impairment in attention. In this case, the child transitions to a 'non-responsive' state where the outcome probabilities become significantly skewed. Given a transition model, we could model this more general case as an MDP with arbitrary, pre-defined states related to the child, the robot, or the child-robot system. In these cases however, the solutions might not have the provable structure of OAssistMe solutions (Theorem 4).

Finally, because of its reliance on minimal assumptions, the family of OAssistMe algorithms is expected to be applicable to settings that go beyond the assistive settings considered in this chapter. Any scenario that requires an agent to make multiple costly attempts at reaching an immediate goal with fixed maximum number of attempts could benefit from our approach. Such scenarios are common for instance in problems related to resource-limited robot control, gaming, or medical decision-making, to name a few.

3.7 Related work

Relevant to our approach are works in the fields of human-agent interaction, Intelligent Tutoring Systems (ITS), healthcare interventions, and robot-assisted autism therapy. We briefly discuss these next.

3.7.1 Probabilistic models for human-agent interaction

While probabilistic models are widely used by agents operating in uncertain environments [129], they seem to be less used in human-interactive contexts. If some human modeling approaches incorporate uncertainty as part of the model [42, 77], planning and adaptation in typical human-computer interaction scenarios mostly do not account for this uncertainty. In the field of HRI however, probabilistic models have gained more interest, both in human-machine teaming settings [104, 38, 91, 148] and mutually adaptive collaborative contexts [112].

In this work, we relied on a simple probabilistic model of the receiver's response to the provider's actions, which introduces uncertainty in the reasoning process of the agent.

3.7.2 Intelligent Tutoring Systems (ITS)

These are computer-based solutions that provide personalized and immediate tools and feedback to learners, with minimal human intervention. There is a very large literature on ITS and a number of approaches consider variations of the personalization problem related to this chapter's goals, according to various context-dependent variables, often with an assumption of partial observability [59, 29]. Grover et al. (2018) specifically frame ITS as a collection of planning problems [71]. Two such problems closest to ours in the ITS literature are the problem of optimal teaching sequence generation [40], and the problem of hint generation [126, 20], which aim at providing tailored context-specific content according to student performance. These problems have mainly been studied in the context of teaching highly structured concepts such as programming or logic proofs [126, 20]. Most state-of-the-art methods rely on a large amount of data, based on algorithms similar to recommender systems, while earlier work tends to be more analytic and model-based [109]. In an agent-based therapy setting, such amount of data is far from being available for a number of reasons, including scarceness of available technologies for special populations, higher-than-normal variability of profiles, and data privacy. As a result, the application of these types of algorithms to therapy contexts is difficult. In this chapter, a relatively small amount of data was needed to be able to estimate model parameters for generating of personalized action sequences. Even though the ITS literature has tackled more complex problems in the past, many of them are not transferable to other domains falling under the provider-receiver interactive paradigm.

The chapter contributed a principled analysis of a simple and general model for certain types of tasks, which we believe may be valuable across a variety of domains. Nevertheless, the ITS field may provide a valuable line of research to accelerate advances in other types of robot-based interventions in the future, especially as more data become available.

3.7.3 Healthcare interventions

Computational approaches to healthcare interventions have been studied both from a technological and decision-making standpoint. From the technological standpoint, Hoey et al. (2013) describe a approach to applying decision-theoretic models to personalized assistive technology for in-home use [76]. Their COACH system (Cognitive Orthosis for Assistive aCtivities in the Home) is closest to the type of tasks we consider in this work, as it focuses on prompting the user to complete a task over a short time frame, using actions with increasing levels of specificity and costs. However, it is unclear how they determine the parameters in their model (e.g., costs). In this work, we favor a more principled approach to investigating how such parameters can be determined from expert and interaction data.

From the decision-making standpoint, there is a body of literature dedicated to decision-theoretic approaches to medical intervention that take into account uncertainty of costly action outcomes. They include MDPs [155, 3] and POMDPs [72], often with a finite time horizon, as is assumed in this chapter. These modeling approaches typically operate over much longer time scales, e.g., the course of a treatment, or maybe even a lifetime. Applications include epidemic control, drug infusion, organ transplantation, screening and treatment, among others [135]. While the algorithms presented in this chapter can be seen as special cases of finite-time MDPs, their structure creates provable properties of optimal solutions (see for example Theorem 4) that are not necessarily valid in more general formulations.

3.8 Summary

This chapter formalized the problem of optimal action selection for a robot acting as a provider, and build on the data collected in Chapter 2. The contributions this chapter makes can be summarized as follows:

1. A mathematical formulation of the optimal action sequence generation problem in a general provider-receiver context and in a multi-trial task with: (1) a scale of actions with known costs and success probabilities, (2) success/failure outcomes at each trial, (3) a horizon corresponding to the maximum allowed number of trials.

2. OAssistMe, a linear-time optimal algorithm based on dynamic programming that solves the above problem.

3. A theoretical analysis of optimal solutions, including proofs of monotonicity and convergence, and constraints on model parameters for suitable algorithm behavior in relation to our application realm.

4. Several extensions of OAssistMe, injecting different assumptions about dependence of action parameters on action history. These extensions are: Trial-Sensitive (TS), Cost-Sensitive (CS), and Repetition-Sensitive (RS) versions of the algorithm.

5. An application of the framework in our robot-assisted ASD therapy setting (from Chapter 2), including a methodology for determining action parameters, namely:
   (1) An online survey with psychologists for determining action costs.
   (2) A probabilistic model of children response to robot actions, based on data collected during a real interaction between a humanoid robot and 10 children with different ASD levels.

In the next two chapter, the assumed robot role will switch from provider to receiver. While the past two chapters focused on the problem of action selection for personalized intervention, we now turn to the problem of embodied simulation for complementing provider training.

Chapter 4

Simulation as Inverse-Assessment

This chapter presents a method for high-level simulation of behavioral responses of receivers (namely, children with ASD) in the context of standardized tasks, accounting for individual differences in receiver profiles. Our method stems from the observation that the simulation process can be seen as the reverse of the assessment process. While assessment maps observed behaviors to a coded profile, simulation uses a profile to generate a realistic set of behaviors consistent with that profile. The goal of our simulation is not to model low-level cognitive processes of children with ASD, but rather to leverage the structure of an existing standardized interactive tool (namely, the ADOS) to simulate a simplistic version of what a real provider-receiver interaction could look like. In the next chapter, we show how our simulation method can be embodied on a robot capable of multi-modal interaction.

The two main algorithmic contributions of this chapter are:

1. *Descriptor-Based Mean Mapping Sampling (DB-MMS)* [14], an algorithm for sampling feature vectors informed by real ADOS data.

2. *Graph-based Behavior Selection (GBS)* [16], an algorithm for selecting compatible behaviors consistent with a given feature vector.

We start by describing some specifics about the ADOS tool, then discuss the two main research questions associated with the simulation problem. The chapter culminates with a demonstration of how the individual contributions integrate into our end-to-end simulator *ADOS-Sim* [15], as well as a discussion on expanding the scope of the research beyond the ADOS specifications. As with the other chapters, we include a brief discussion of related work before summarizing the main takeaways.

4.1 Background and research questions

We first provide some specifics on the structure of the ADOS, then identify the main research questions needed for the development of our simulation approach.

4.1.1 Structure of the ADOS tool

The ADOS comprises five modules suitable for different language abilities and/or ages. Module 1 (Pre-verbal/Single Words) remains the main module used by therapists for an initial assessment of children from 31 months up to 14 years of age. As in Chapter 2, we focus on this particular module in this chapter, but our methods can be directly applied to other modules as they possess a very similar structure.

Tasks

The ADOS contains 10 standardized tasks ('activities' in ADOS terms) listed in Table 4.1. These tasks have varying degrees of structure. They range from rather unstructured activities such as PLAY, where the child is left to freely play in the room, to very structured activities such as NAME, where the therapist calls the child's name at very specific degrees of explicitness and observes the child's response. Note that the JATT and NAME tasks were used in Chapter 2 and are also used in Chapter 5 along with the SNACK task. In a typical ADOS session, the therapist performs the activities and records behaviors of interest throughout the session. After completing all tasks, the therapist codes the observed behaviors as a set of values on different features, as explained next.

Features

There are a total of 34 integer-valued ADOS features ('codes' in ADOS terms) capturing different, usually exclusive, behavior types. They cover five behavioral categories, namely 'Language and Communication', 'Reciprocal Social Interaction', 'Play', 'Stereotyped Behaviors and Restricted Interests', and 'Other Abnormal Behaviors'. Features have value ranges starting at 0 and going up to 2–4, depending on the feature. A value of 0

Table 4.1 List of the 10 standardized tasks of the ADOS[101], along with labels used to refer to them in this thesis, and short descriptions.

Task name	Label	Description
Free Play	PLAY	Leave the child to play freely in the room with some available toys
Calling Name	NAME	Call the child's name in different ways until a satisfactory response is observed
Joint Attention	JATT	Call the child's attention to a remote-controlled toy in different ways until a satisfactory response is observed
Bubble Play	BUB	Blow soap bubbles to engage the child
Anticipation of Routine with Objects	OBJ	Establish a routine with a cause-and-effect toy such as a balloon and attempt to get the child to show signs of anticipation
Responsive Social Smile	SMILE	Direct a smile at the child in different ways and observe their response
Anticipation of a Social Routine	SOC	Play peekaboo, tickle or swing the child repeatedly
Functional and Symbolic Imitation	IMIT	Play an imitation game with some objects and placeholders for those objects
Birthday Party	BDAY	Play a make-believe game with a doll who is having a birthday party
Snack Preference	SNACK	Present the child with different snacks and ask them for their preference

typically represents absence or minimal impairment while higher values signify higher degrees of impairment. The coding scheme specifies clear coding guidelines for every feature. Some special values are also reserved for unusual cases such as non-applicability or inability to judge based on child responses. For the purposes of the scoring algorithm used for diagnosis (described in the next subsection), only 14 out the 34 features are considered [68]. These features are shown in Table 4.2. For a complete list of ADOS features, we refer the reader to Appendix A.

Through its use of numerical features to code behaviors, the ADOS effectively defines a *feature space* for ASD. Every point in this space is a *feature vector* that represents an individual with a unique set of ASD characteristics. We refer to a feature vector as $\mathbf{f} = \langle f_1, \ldots, f_M \rangle$. For the purposes of ADOS-Sim, M is set to 14. For data analysis and experimental purposes, we sometimes use a different value for M.

Scoring algorithm

The purpose of the scoring algorithm is to convert a feature vector into a single *total score*, which we denote by Σ, ranging from 0 to 28. As mentioned previously, the scoring algorithm only considers 14 out of 34 ADOS features. The list of features varies slightly depending on language ability (as defined in the next subsection). This observation is captured by the last two rows of Table 4.2. As a pre-processing step, feature values are all converted to the 0–2 range before they are summed up to produce the total score $\Sigma = \sum_{i=1}^{M} f_i$.

The total score can be further broken down into three *subtotals* for *'Communication' (Comm.), 'Reciprocal Social Interaction' (Soc.), and 'Restricted and Repetitive Behaviors' (RRB)*. From the total score, a *severity* value between 1 and 10 is produced using a conversion table (not included) that accounts for the child's age and language ability.

Descriptors

Unlike features that encode information about specific aspects of behaviors, we define descriptors as high-level variables that are relevant enough to be included as part of the child profile. We consider the following three descriptors:

- *Age* — We consider ages ranging from 2 to 14 years, consistent with the scope of the ADOS Module 1.

Table 4.2 List of the ADOS features used for the scoring algorithm [101]. Depending on the language ability, this list differs slightly as shown in the last two columns. ✓ means included and ✗ means excluded.

Subtotal group	**Feature**	**Feature label** ($\hat{f}$)	**Language ability** 'none'	'some'
Communication (Comm.)	Frequency of Vocalization Directed to Others	SVOC	✓	✓
	Pointing	PNT	✗	✓
	Gestures	GES	✓	✓
Reciprocal social interaction (Soc.)	Unusual Eye Contact	EYE	✓	✓
	Facial Expressions Directed to Others	EXPO	✓	✓
	Integration of Gaze [etc.] During Social Overtures	GAZE	✓	✓
	Shared Enjoyment in Interaction	ENJ	✓	✓
	Showing	SHO	✓	✓
	Spontaneous Initiation of Joint Attention	IJA	✓	✓
	Response to Joint Attention	RJA	✓	✗
	Quality of Social Overtures	QSOV	✓	✓
Restricted and Repetitive Behaviors (RRB)	Intonation of Vocalizations or Verbalizations	IN	✓	✗
	Stereotyped/Idiosyncratic Use of Words or Phrases	STER	✗	✓
	Unusual Sensory Interest in Play Material/Person	SINT	✓	✓
	Hand and Finger and Other Complex Mannerisms	MAN	✓	✓
	Unusually Repetitive Interests or Stereotyped Beh.	RINT	✓	✓

- *Language ability* — This descriptor is assessed by the therapist at the end of an ADOS session. It is assigned to one of two values: *'none'* for children using few to no words, and *'some'* for children using some words in their speech.
- *Severity* — This integer-valued descriptor ranges between 1 and 10, and is computed as described in the previous subsection.

The first two descriptors are used to convert the total score to a severity value. Therefore, we assume that these three descriptors can be seen as independent variables that provide a high-level characterization of the child. In our ADOS-Sim simulator, these descriptors are used as inputs specified by the user. Hence, the simulator is able to account for the full range of severities, ages, and language abilities covered by the ADOS.

Behaviors

Every feature value is accompanied by a list of several common behaviors that fall under that value. In this chapter, a behavior is represented as an English description of how the child uses its *behavioral channels* (e.g., speech, gaze, body motion, emotional expression) either spontaneously or as a response to a specific action performed during one or more tasks. In Chapter 5, we design matching robot behaviors for a few features, enabling an embodied visualization of these behavior descriptions. Behaviors can be very specific, e.g., "Directs facial expressions to examiner to express puzzlement", or rather general, e.g., "Appropriately gazes while communicating". A behavior associated with value v on feature $\hat{f}$ is denoted by $b(\hat{f}, v)$. As there is typically more than one behavior that falls under a single (f, v) pair, we index behaviors as $b(\hat{f}, v)_i$.

4.1.2 Research questions

Figure 4.1 summarizes the relation between the ADOS concepts introduced, and shows a high-level overview of how the assessment process will be reversed in this chapter to achieve the desired simulation goals.

There are two main challenges associated with reversing the ADOS pipeline. The first concerns the notion of *realistic* feature vector sampling. For example, for a total score $\Sigma = 14$, there are more than 600,000 possible feature vectors summing up to Σ. How do we know which ones are more realistic than others? Uniformly sampling these

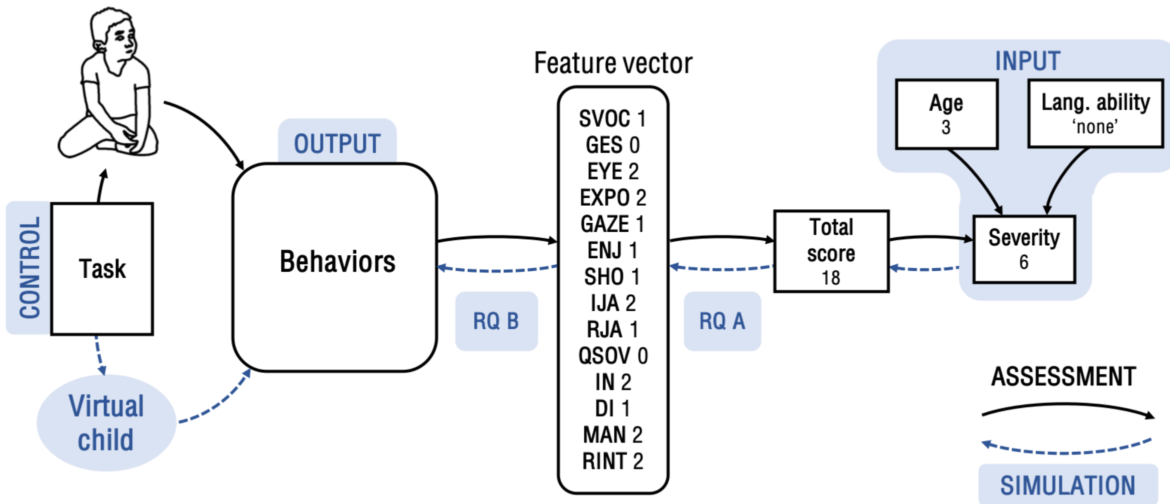

Figure 4.1 Simulation as inverse-assessment: we reverse the ADOS process to achieve and end-to-end simulation. Numbers in small fonts represent an example that the reader can track. Input, output, and control elements used in ADOS-Sim are shown as labels.

feature vectors would not be appropriate as it does not capture the interdependence between features, and hence we need to look for an alternative. The second challenge concerns the notion of *incompatible* behaviors between features. Based on how behaviors are described in the ADOS manual, there are several cases where two behaviors belonging to two different features can carry contradicting information, for example:

> Example 1 — "Exhibits an odd cry and **no other vocalizations**" (feature IN) and "**Vocalizes** to be friendly" (feature SVOC).
>
> Example 2 — "Uses **poorly modulated eye contact** to initiate social interaction" (feature EYE) and "Uses **eye contact** to get help" (feature GAZE).

Because of such cases of seemingly contradicting information between behaviors from different features, the simulator must take into account compatibility between behaviors when selecting them.

In light of these challenges, the two main research questions we investigate in this chapter are:

RQ A — Given descriptors, such as age, language ability and ASD severity, how to sample realistic feature vectors?

RQ B — Given a feature vector, how to select compatible behaviors within each task?

For RQ A, we use real data to inform our sampling method, and contribute the DB-MMS algorithm. For RQ B, we use a graph representation of behaviors that encodes compatibility of behaviors. Using this representation, we contribute the GBS algorithm, which selects behaviors while respecting compatibility constraints within each task.

After tackling these two questions separately, we end the chapter by connecting all components of the ADOS-Sim simulator and showing how the methods developed in RQ A and RQ B fit into its architecture.

4.2 Feature vector sampling (RQ A)

Our goal is to sample a feature vector given constraints imposed by descriptors. In ADOS-Sim, the descriptors are in the form of a triplet (age, language ability, severity).

The first step is to map the given descriptors to a total score. This step is trivial since it is directly given by a conversion table present as part of the ADOS manual. This table converts a given triplet (age, language ability, total score range) to a single severity value. Reversing this conversion gives us, for a given descriptor triplet (age, language ability, severity), a range for the total score. The width of this range in the available table varies between 0 and 7 depending on the triplet combination. We then uniformly sample an integer value in that range as our total score Σ.

The second step is to set individual feature values given the total score constraint. In other words, we are looking for a method to sample a feature vector such that Σ is equal to a specified value. As mentioned previously, some vectors will be unlikely to occur because we do not expect to have independence between feature values f_i. We use real data to verify and make use of this hypothesis, as explained in the following subsections.

The rest of this section is organized as follows. We first present the dataset used to inform our sampling method and analyze the distribution of the data using dimensionality reduction techniques. We then present the DB-MMS algorithm, which samples feature vectors by considering pairwise correlations between features, estimated from the real data, while respecting descriptor constraints.

4.2.1 Dataset description

The dataset consists of the full ADOS Module 1 score set (values on all ADOS features) of children suspected of having an ASD ($n = 279$). The data came from two sources: 212 data points were obtained from the National Institute of Mental Health Data Archive (NDA)[1] and 67 data points were obtained from the Child Development Center at the Garcia de Orta Hospital in Almada, Portugal[2]. Ages range between 2 and 19 years ($\mu = 5.64$, $\sigma = 3.30$). Part of the dataset does not have gender information, but for the 147 data points that have it, the male-to-female ratio is 38:49.

This type of data presents some challenges, outlined below:

- Data are *discrete* — Non-continuous data make it harder to generate synthetic feature vectors that are consistent with the real data. For example, while sampling synthetic data points according to a correlation model is straightforward for real-valued Gaussian feature values, sampling correlated discrete data it is not. The DB-MMS algorithm addresses this challenge and is specifically designed to sample correlated discrete data.
- Data are *ordinal* — Traditional parametric methods might not be suitable for this type of data. As a result, we use a non-parametric measure of correlation between feature pairs.
- Data are *noisy* — Although the ADOS is a standardized tool, it is know to have some level of subjectivity in its coding scheme, as different therapists may assign different values for a same set of observations during an session [154]. We keep the presence of noise in mind during our data analysis.
- Data are sometimes *incomplete* — The dataset has missing entries (`NaN`s) in some features for some of the subjects (28 of 8091 entries were missing, mostly for feature IN). We replaced `NaN`s, in addition to other special coded values, with randomly

[1]The NDA is a US-based collaborative informatics system that serves as a national resource to support and accelerate research across several scientific disciplines, including autism research.

[2]These ADOS scores are part of a database kept for statistical purposes. All data are anonymous; only age and gender were collected from the sample for demographic characterization.

sampled values in the allowed range. Even though more advanced methods of dealing with missing data such as matrix completion [33] could be used, this simple method ensured that the correlation structure of the data, the basis for our feature vector sampling algorithm, was maintained after pre-processing, at the cost of contributing to some additional noise.

- Data is *heterogeneous* — The dataset includes data from both the original version of the tool and the second version ADOS-2 (which we have been using so far). The main differences between the newer and the older versions are:

  - The coding rules for one of the features were slightly revised and 4 out of 29 features had their range changed from a 3-point to a 4-point scale. Because values greater than 2 all count the same in the scoring algorithm, we did not transform the data.
  - Five features were added, but none of these are part of the scoring algorithm. In the rest of this chapter, we neglected any features that were not common to both versions, resulting in a total of $M = 29$ features.
  - The scoring algorithm was slightly revised, for added robustness in diagnosis [68]. We use the more recent algorithm to compute total scores as it is an improvement over the first one.

4.2.2 Data analysis

To get better insight into the data, we first analyze its distribution across the full feature space ($M = 29$). We present two methods for visualizing the feature vectors of the dataset using a lower-dimensional, human-readable representation. The first one uses an unsupervised learning approach consisting of a Self-Organizing Map (SOM) [89] to map the data into a two-dimensional space. The second one uses the three-dimensional space formed by the ADOS subtotals ('Comm.', 'Soc.', and 'RRB') directly. These visualizations are useful to get an idea of the data distribution, but also to inform the generation of new descriptors based on the data (Section 4.5.1).

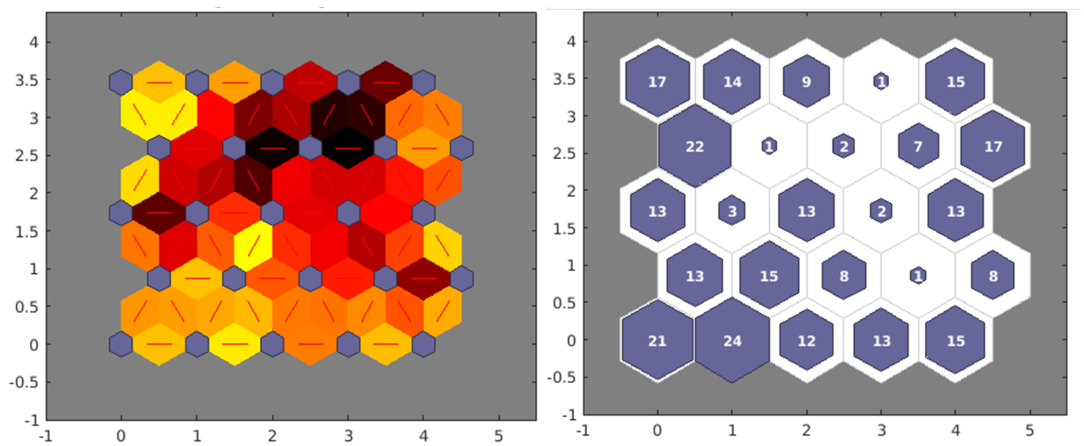

Figure 4.2 2D visualization of the dataset ($M = 29$) using an SOM. *Left*: U-Matrix showing distance between neighboring neurons. The darker the color, the more separated the connected neurons. *Right*: Sample hit histogram showing the number of data points mapped to each neuron. *SOM parameters*: #(epochs) for training:1,000; distance type:link distance; initial input space covering:100; initial neighborhood size:3.

Dimensionality reduction using the SOM method (2D)

A Self-Organizing Map (SOM) is a neural network that learns, in an unsupervised way, an alternative, low-dimensional, representation of high-dimensional data in the form of a 2D map consisting of interconnected neurons preserving the topology of the original data [89]. We trained on the dataset an SOM consisting of 25 neurons connected in a 5-by-5 hexagonal map. With each neuron, there is an associated position in the map space, as well as an associated weight learned by the training algorithm.

A common visualization method of an SOM is through the unified distance matrix (U-matrix) [151], which computes the distance between a neuron and its neighbors in the map space. The left part of Figure 4.2 shows the U-matrix for our trained SOM, where brighter regions correspond to more clustered regions and darker regions correspond to lower-density regions. Our U-matrix suggests that there are no clearly separated clusters in the data, but rather some low-density regions in the feature space.

The trained network maps input feature vectors to the closest neuron in the map space. The right part of Figure 4.2 shows a histogram of the number of data points being mapped to each neuron for the dataset. This plot confirms the intuition we got from the U-matrix that there are low-density regions in the dataset rather than clearly separated clusters.

As a final note, we justify our choice of the main SOM hyperparameter, namely its size. Even though no systematic validation of our size choice was performed, it was chosen as a result of experimenting with different sizes, as a tradeoff between overfitting and generalization power, especially in relation to the resulting sample hit histogram, which gives us an idea of the probability distribution across the map. It is worth mentioning that no clear cluster separation was found even with larger SOM sizes.

Dimensionality reduction using ADOS subtotals (3D)

The three ADOS subtotals, as introduced in Section 4.1.1 are Comm. (range 0–6), Soc. (range 0–16), and RRB (range 0–10). Figure 4.3 shows the data points in the 3D space formed by the subtotals, where each axis corresponds to one subtotal. Unlike the previous dimensionality reduction method, this method only considers the features included in the scoring algorithm. Based on the ADOS, we further cluster the data into the following four severity classes according to the severity value (1–10) for each subject:

- Severity 1–2 → Minimal to no evidence
- Severity 3–4 → Low
- Severity 5–7 → Moderate
- Severity 8–10 → High

The general observation is that, consistent with our SOM analysis, the data points do not form clearly separated clusters, but rather present some low density regions. Now that we got a sense of how the data is distributed in the feature space, we look at correlations between features to inform our feature vector sampling method.

Correlation analysis

In a second step, we performed a correlation analysis, reported in Figure 4.4. We only considered pairwise correlations between features, and ignored higher-order correlations. As our correlation metric, we use Spearman's rank correlation coefficient, which is a non-parametric measure (real number in $[-1; 1]$) capturing how well a monotonic function can be used to describe the relation between two random variables [45]. The sign indicates whether this function is increasing or decreasing. It is a well suited measure of correlation when dealing with ordinal variables like our features.

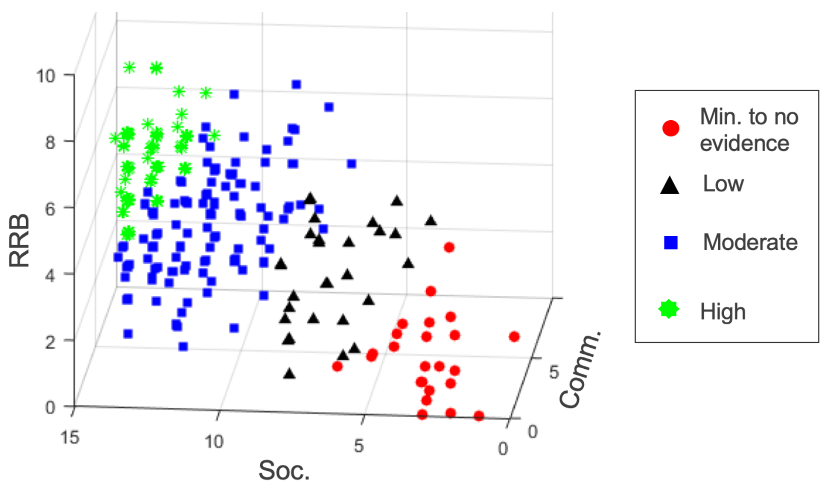

Figure 4.3 Distribution of data points in the 3D space formed by the ADOS subtotals, visualized according to different ASD severity classes ($M = 16$). Overlapping points were slightly disturbed for better visualization.

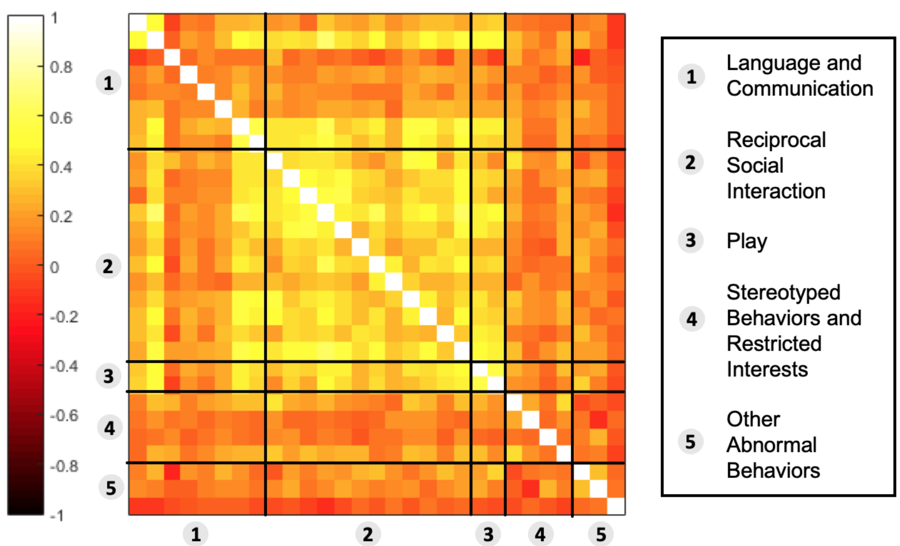

Figure 4.4 Spearman correlation matrix for the dataset ($M = 29$). Rows and columns represent the features in the order of Table A.3, grouped according to the five behavioral categories shown on the right.

Our results show that most features are positively correlated with statistical significance (according to a t-statistic with 0.05 p-value threshold). This result is not surprising given that neurological, developmental, and genetic causal relationships have been found to explain a large set of different behaviors in subjects with ASD [36]. On the other hand, some pairs of features show a low or even slightly negative correlation, especially in the 'Stereotyped Behaviors and Restricted Interests', and 'Other Abnormal Behaviors' categories. This correlation analysis will be used as a basis for our sampling algorithm described in the next subsection.

4.2.3 Descriptor-Based Mean Mapping Sampling (DB-MMS) algorithm

The DB-MMS algorithm samples synthetic feature vectors that (1) preserve the pairwise correlation between feature values for large number of samples, and (2) satisfy a logical constraint (True/False) involving feature and descriptor values. The algorithm is an extension of the mean mapping method developed by Kaiser et al. (2011) [84].

One may wonder why a sampling algorithm is needed in the first place when one could simply select actual feature vectors directly from the database. The first advantage our method offers is that it bypasses any privacy issues that may be associated with the data used. Even if anonymized, healthcare data are typically subject to very stringent privacy guidelines. Therefore using the correlation information, as opposed to a database of real examples, further protects the privacy of individuals whose data are being used. The second advantage is that it allows to generalize from real examples, allowing for the generation of a more varied set of realistic profiles that are not present in the real data.

Sampling correlated feature values

Our first aim is to generate synthetic data such that correlations between the features are maintained. Since the features are discrete and ordinal, such a task is non-trivial. There exist methods to sample ordinal correlated data such as the Gaussian copula [105], the binary conversion, and the mean mapping methods [84]. Out of these methods, mean mapping gave the best results for the data at hand. The method takes as an input the target correlation matrix and the marginals for each feature, and performs the following steps [84]:

1. For the given marginals, compute the quantiles assuming an underlying Gaussian model for each feature.

2. Estimate a corresponding correlation matrix in continuous space, where ordinal variables are replaced with underlying Gaussian variables. This step involves interpolating a function over a regular grid of computed probabilities to estimate correlation coefficients.
3. Sample normal data according to the estimated corresponding correlation matrix.
4. Cut the samples according to the computed quantiles to get back ordinal data.

Figure 4.5 shows the result of applying the base mean mapping sampling method to the dataset. For simplicity, we show the 12 features common to the scoring of both 'no words' and 'some words' language abilities to avoid splitting the dataset into two and losing statistical power. We also set the target feature marginals to uniform distributions. As the number of samples increases, the sample correlation matrix converges to the target matrix from the dataset, as expected. Figure 4.6 further shows the distribution of total scores for the sampled feature vectors. The distribution is almost uniform, which we attribute primarily to the target uniform marginals we enforced.

Incorporating constraints from descriptors

Our second aim is to make sure that the sampled feature vector is consistent with the specification of the descriptors. In our case, descriptors translate into a range of total scores, so we want to constrain the sum of feature values to fall within that range. In general, descriptors can impose any arbitrary logical constraint on the feature vector. We denote this logical constraint by $\psi(\mathbf{f}, \mathbf{D})$, where $\mathbf{f}$ is an arbitrary feature vector, $\mathbf{D}$ is an arbitrary set of descriptors, and ψ is a Boolean function (True/False).

To incorporate the constraint into the sampling algorithm, we use rejection sampling. In other words, the generated samples that satisfy $\psi(\mathbf{f}, \mathbf{D})$ are accepted, and the ones that do not are rejected. This method does not depend on the choice of descriptor or feature choice, as long as the constraints imposed by the descriptors can be expressed as a Boolean statement.

Algorithm 5 shows a pseudocode of the DB-MMS algorithm. Parameters of the mean mapping method `params`, including the corresponding correlation matrix and quantiles, only need to be computed once before starting the sampling process. The distribution of total scores in the unconstrained algorithm shown in Figure 4.6 suggests that the amount of computation needed to generate a feature vector for a given total score through

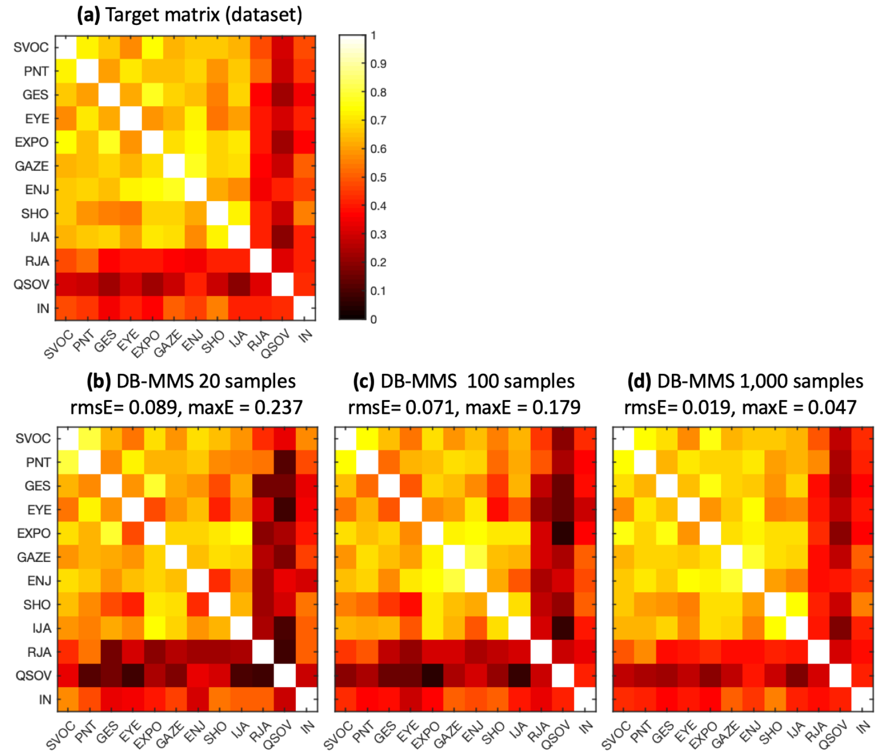

Figure 4.5 Correlation matrices generated by the mean sampling method ($M = 12$), along with root-mean-square and maximum absolute error with respect to target. Note adjusted color range for better visibility as compared to Figure 4.4, since all values are positive.

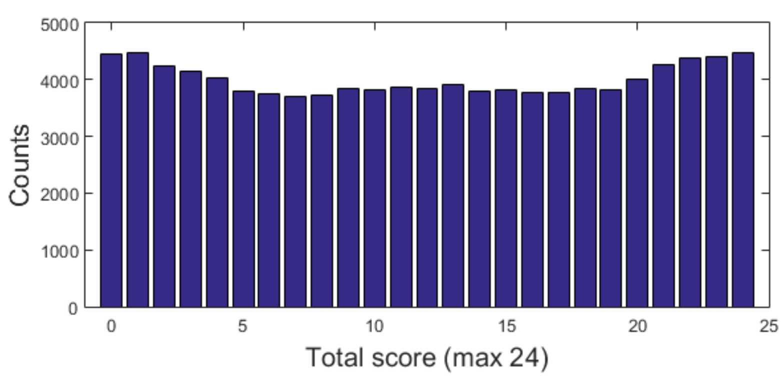

Figure 4.6 Histogram of total scores for 100,000 sampled feature vectors ($M = 12$, consistent with Figure 4.5, hence the maximum total score of 24).

Algorithm 5 Pseudocode of the DB-MMS algorithm that samples m feature vectors according to the specifications of descriptors $\mathbf{D}$, and targeting correlation matrix `corr` and feature marginals `marg`. Details of functions in line 4 and 6 can be found in [84].

1: **procedure** DB-MMS(m,$\mathbf{D}$,corr,marg)
2: $\quad \mathbf{F} \leftarrow \emptyset$
3: $\quad i \leftarrow 0$
4: $\quad$ params $\leftarrow$ MeanMappingParams(corr,marg) $\triangleright$ Steps 1 and 2 in text
5: $\quad$ **while** $i < m$ **do**
6: $\qquad \mathbf{f} \leftarrow$ MeanMappingSample(params, $\mathcal{U}$)
$\triangleright$ Steps 3 and 4 in text; $\mathcal{U}$ is the uniform distribution over the feature value range
7: $\qquad$ **if** $\psi(\mathbf{f}, \mathbf{D})$ **then** $\triangleright$ e.g., $\sum_i^M f_i = \Sigma$ or $\Sigma_{\min} \leq \sum_i^M f_i \leq \Sigma_{\max}$
8: $\qquad\quad \mathbf{F} \leftarrow \mathbf{F} \cup \{\mathbf{f}\}$
9: $\qquad\quad i \leftarrow i + 1$
10: $\quad$ **return** $\mathbf{F}$

rejection sampling does not significantly rely on the value of that score. The DB-MMS algorithm was implemented in R based on the `orddata`[3] package.

4.3 Behavior selection (RQ B)

We now tackle our second research question, concerned with selecting individual behaviors within tasks, based on a given feature vector (e.g., sampled by DB-MMS).

The questions we seek to answer in this section are:

- Which features are relevant to which task?
- How to encode the existence of incompatibilities between behaviors from different features (as discussed in Section 4.1.2)?
- How to select compatible behaviors once the previous two questions have been answered?

4.3.1 Behavior database creation

Based on the coding instructions of the ADOS manual, we manually populated a database of behaviors to be used in our simulation approach. The manual lists the most common

[3] https://r-forge.r-project.org/R/?group_id=708

behaviors, with varying degrees of specificity, that would fall under a given value for each feature. For every feature $\hat{f}$ and value $v = 0, 1, 2$, we extracted a set of behaviors $b(\hat{f}, v)_i$. The number of behaviors that fall under the same $(\hat{f}, v)$ pair ranges from 1 to 8 behaviors, with varying degrees of similarity between behaviors. This extraction process resulted in a database with a total of 123 behaviors across the 16 features of Table 4.2.

Example of a database entry for feature PNT:

- **PNT=0**:
  $b(\mathrm{PNT}, 0)_1$: "Points with index finger to show visually directed referencing".

- **PNT=1**:
  $b(\mathrm{PNT}, 1)_1$: "Produces an approximation of pointing".
  $b(\mathrm{PNT}, 1)_{2-7}$: "(Gazes)/(vocalizes) while (touching object)/(pointing to a person)/(pointing to self)" (all combinations).

- **PNT=2**:
  $b(\mathrm{PNT}, 2)_{1-2}$: "Points when (close to)/(touching) object only, and with no gaze or vocalization".
  $b(\mathrm{PNT}, 2)_3$: "Does not point".

4.3.2 Identifying task-relevant features

The ADOS requires feature coding to happen after all tasks are completed, hence there is no clear list of features that are relevant in every task. In fact, some features seem to be totally irrelevant to some tasks, while others seem to be relevant for multiple tasks. Because our simulation discriminates between different tasks, it is important to identify which features are more likely to be exhibited by the child in which task. Despite the lack of a list of task-relevant features, the ADOS manual does provide some observation guidelines within each task. These guidelines help the therapist determine behaviors they should focus their attention on for every task.

Based on these guidelines as well as the nature of the task, we identified a set of task-relevant features, summarized in Table 4.3. Relevant features capture the types of behaviors that are expected to be exhibited in – or are of special importance for – a particular task. For example, in task JATT, the feature 'Spontaneous Initiation of

Table 4.3 Features identified as relevant for each task based on the ADOS manual (includes all features in Table 4.2).

Task	**Relevant features**
Free Play	SVOC,STER,GES,ENJ,SHO,PNT,IJA, EXPO,SINT,MAN,RINT
Calling Name	EYE,GAZE,SVOC,STER
Joint Attention	RJA,EYE,SVOC,STER,ENJ,IJA,QSOV
Bubble Play	EXPO,IJA,ENJ,SINT,MAN,RINT
Anticipation of a Routine with Objects	IJA,ENJ,MAN,RINT
Responsive Social Smile	EXPO
Anticipation of a Social Routine	GAZE,EYE,EXPO,SVOC,STER,GES,ENJ
Functional and Symbolic Imitation	GAZE,ENJ
Birthday Party	RINT,ENJ,QSOV,EYE,EXPO,GAZE
Snack Preference	EYE,GES,EXPO,IN,QSOV

Joint Attention' (IJA) is relevant, but the feature 'Unusual Sensory Interest in Play Material/Person' (SINT) is not. A note of caution is that this assignment has not been validated with ADOS specialists, but is simply seen as an improvement over a completely task-agnostic simulation.

4.3.3 Graph representation of behaviors

In order to capture compatibility between behaviors, we introduce *behavior compatibility graphs*, where vertices represent behaviors and edges represent pairwise compatibility. This graph representation of behaviors turns the behavior selection problem into a subgraph selection problem.

While a single behavior graph could be built for all features, it could result in some issues. Being too restrictive, it would not produce a rich enough set of behaviors. In fact, while behaviors are expected to be compatible within each task, children with ASD are often unpredictable and it would not be unlikely for them to exhibit incompatible behaviors across different tasks. For these reasons, we choose to keep separate compatibility graphs for each task, thereby aiming at ensuring *within-task compatibility of behaviors*. Because we select a single behavior for each feature at every simulated run of task, no edges are allowed for behaviors belonging to the same feature. As a result, behavior compatibility graphs are k-partite graphs, where k is the number of relevant features.

Figure 4.7 shows a sample behavior compatibility graph for task NAME. The graph includes the four features relevant to that task according to Table 4.3, with sample values assigned: SVOC=2, GAZE=1, EYE=0, and STER=0. The way we identify behavior compatibility or lack thereof is described next.

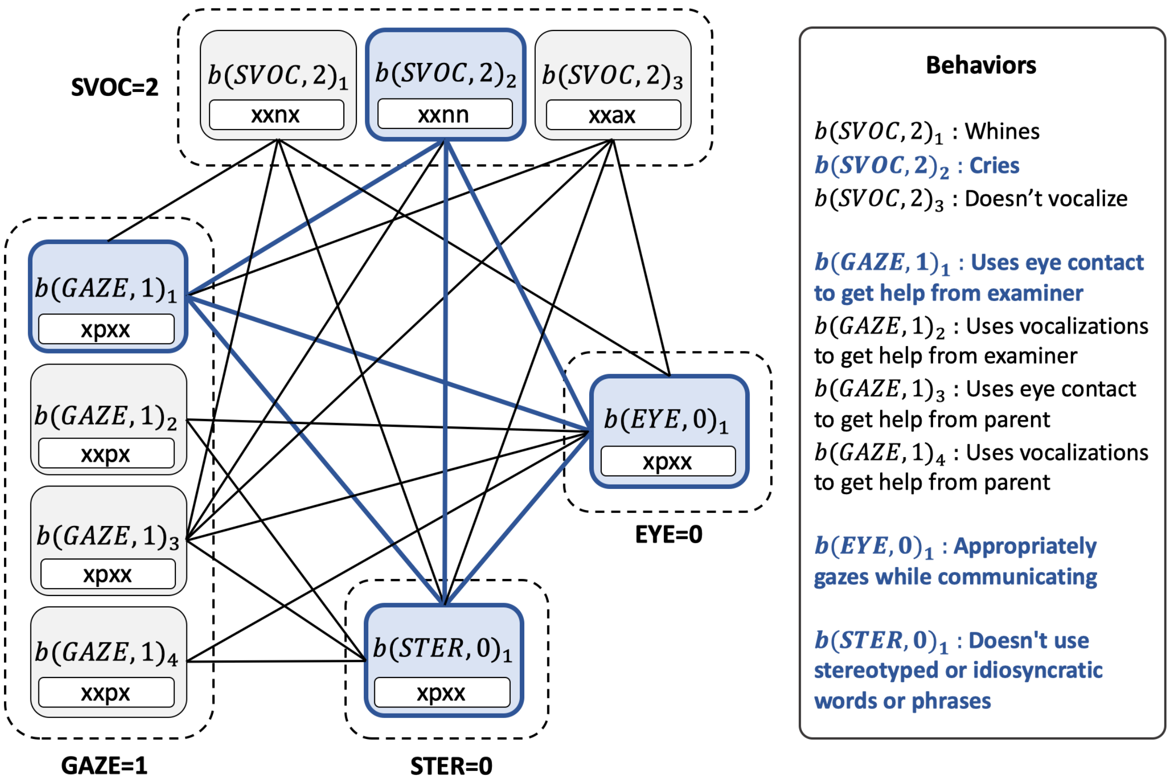

Figure 4.7 Sample behavior compatibility graph (4-partite) for task NAME. *Emphasized*: choice of pairwise compatible behaviors as a fully connected subgraph. Behavioral channel vectors listed are of the form: ⟨Body motion, Gaze, Speech, Emotional expression⟩.

4.3.4 Building the behavior compatibility graphs

Due to the large number of behaviors, we cannot possibly define the compatibility of every behavior pair by hand. As an approximate solution for specifying these compatibilities, we introduce *behavioral channels*. These correspond to different expression modalities that can co-exist independently. In addition to simplifying the specification of compatibility, decomposing individual behaviors into channels simplifies the mapping of multi-modal behaviors to expressive channels on an artificial agent such as a robot. In fact, while this chapter only considers English descriptions of behaviors, in the next chapter behaviors will be understood as embodied on a robotic agent.

The four behavioral channels we consider are: 'Body motion', 'Gaze', 'Speech', and 'Emotional expression'. However, the choice of channels is not restricted and could differ according to the intended embodiment of the simulator, or to the simulation purpose. On each channel, we define four possible values:

- 'x': no mention of specific behavioral content on channel.
- 'a': specified absence of behavioral content on channel (e.g., no speech).
- 'p': presence of positive behavioral content on channel (e.g., smile).
- 'n': presence of negative behavioral content on channel (e.g., crying).

To each behavior, we associate a *behavioral channel vector* consisting of four values, one for each channel. Sample behavioral channel vectors are listed under each vertex in Figure 4.7. For example, behavior $b(SVOC, 2)_2$ (crying) has negative behavioral content on both the 'Speech' channel (which includes vocalizations, such as those related to crying) and the 'Emotional expression' channel. The other two behavioral channels are unspecified, which results in the behavioral channel vector $\langle$xxnn$\rangle$.

Compatibilities are determined according to valid combinations of the behavioral channel values across behaviors. In particular, $\{n, p\}$, $\{n, a\}$, and $\{p, a\}$ are considered incompatible, while all other combinations are considered compatible. For example, $b(SVOC, 2)_3$ is compatible with $b(GAZE, 1)_1$, but not with $b(GAZE, 1)_2$ due to the conflicting values 'p' and 'n' in the 'Speech' channel. This method reduces the number of needed annotations to make from quadratic to linear in the number of behaviors in the database. It also allows for more flexible and compact specifications for criteria of compatibility by specifying a small set of compatible or incompatible combinations of behavioral channel values.

While our assignment of behavioral channel values for our extracted behavior database was obvious for most behaviors, some cases were more fuzzy. First, very specific behaviors that only appear in a single feature (e.g., absence of pointing, showing, sensory interests, etc.) were coded as 'x' when specified as absent. Also, some examples where the coding was not straightforward include the expression of puzzlement and skepticism in the EXPO feature, coded as negative emotional expressions. Furthermore, the specified absence of a clearly positive behavior was coded as negative (e.g., 'no expressed pleasure' was coded as 'n' on the 'Emotional expression' channel).

4.3.5 Graph-based Behavior Selection (GBS) algorithm

The GBS algorithm takes as an input a task and a feature vector and returns a combination of behaviors (one for each relevant feature) that are all pairwise compatible. To do so, it starts by building the behavior compatibility graph corresponding to the assigned feature values, adding an edge between two vertices whenever the compatibility conditions are met on all channels. Once the graph is built, it finds a maximum fully connected subgraph (max-clique), an example of which is emphasized in blue in Figure 4.7. Because the graph is k-partite a max-clique will cover all features with no repetition of the same feature. A pseudocode for GBS is shown in Algorithm 6, which we implemented in Python programming language. For the example of Figure 4.7, calling GBS(NAME) returns the set of behaviors $\{b(SVOC, 2)_2, b(GAZE, 1)_1, b(EYE, 0)_1, b(STER, 0)_1\}$.

Algorithm 6 Pseudocode of the GBS algorithm that selects compatible combinations of behaviors for each task.

procedure GBS(task)
    $\mathbf{F}_{\text{relevant}} \leftarrow$ GetRelevantFeats(task)
    $\mathbf{B} \leftarrow$ GetBehaviors($\mathbf{F}_{\text{relevant}}$)
    $\mathbf{E} \leftarrow \emptyset$
    for all $(b, b') \in \mathbf{B}^2$ s.t. b and b' from different features **do**
        comp $\leftarrow$ TRUE ▷ Compatibility indicator
        for all channel $\in$ channels **do**
            if $\neg$ IsCompatible(b,b',channel) **then**
                ▷ Checks for forbidden combinations $\{n, p\}$,$\{n, a\}$,$\{p, a\}$
                comp $\leftarrow$ FALSE
                break
        if comp **then**
            $\mathbf{E} \leftarrow \mathbf{E} \cup \{(b, b')\}$
    $\mathbf{G} \leftarrow (\mathbf{B}, \mathbf{E})$ ▷ Behavior compatibility graph
    $\mathbf{B} \leftarrow$ FindMaxClique($\mathbf{G}$)
    return B

Note that the max-clique problem is known to be NP-hard [26]. However, because we are considering separate graphs for every task, our graphs are relatively small resulting in manageable running times. If we had considered a single graph for all features, the computational complexity would have been problematic.

4.4 ADOS-Sim simulator

We now show how the different components investigated in RQ A and RQ B integrate into the full ADOS-Sim simulator, whose architecture is shown in Figure 4.8. The input descriptors first get translated into a total score range using the ADOS conversion table. In this range, a single total score is randomly selected. The DB-MMS algorithm then samples a feature vector whose sum matches the specified total score, using the correlation matrix of the dataset. Finally, the GBS algorithm selects behaviors from the database according to the list of task-relevant features from Table 4.3. We implemented the full simulator, including a simple graphical user interface, in Python, with dependency on R. A snapshot of the user interface is shown in Figure 4.9.

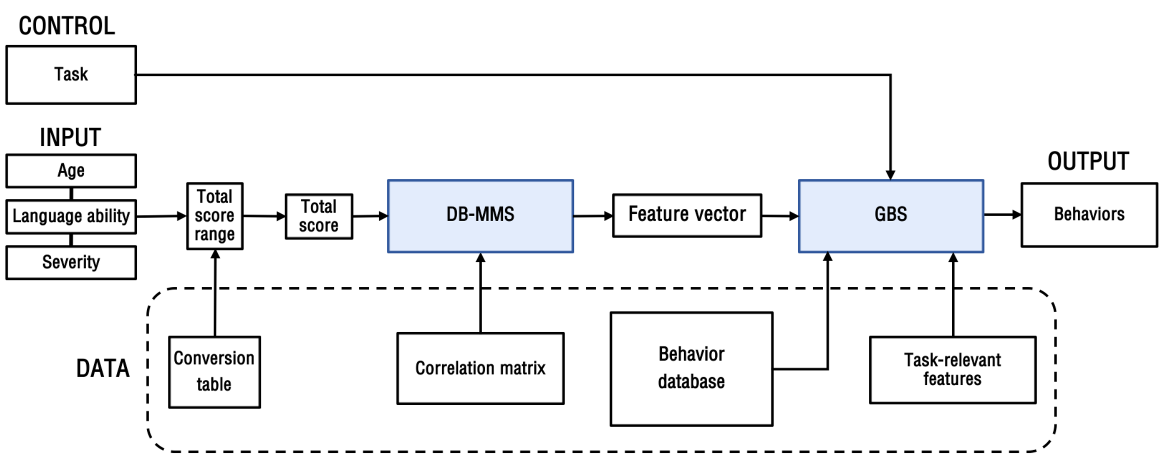

Figure 4.8 Overview of the ADOS-Sim architecture. Algorithm blocks are shown in filled blue to differentiate them from variable blocks.

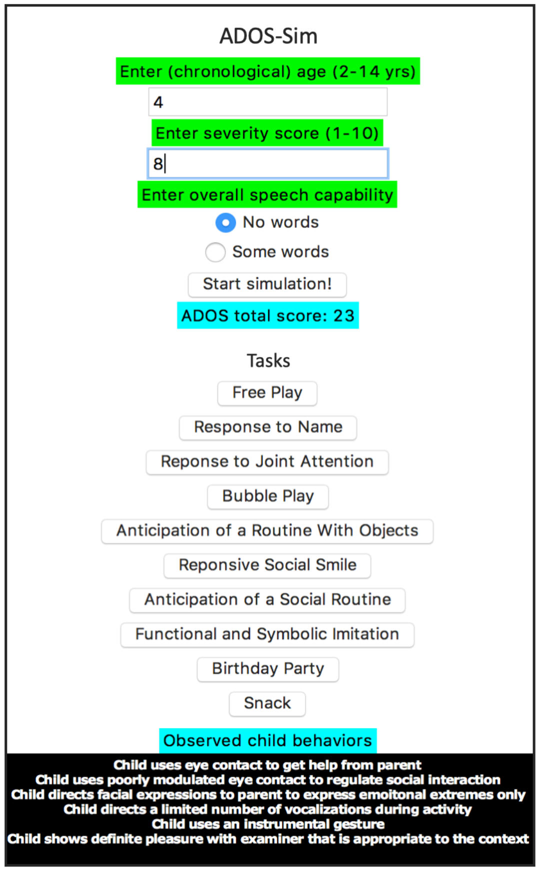

Figure 4.9 Snapshot of the ADOS-Sim user interface.

4.5 Beyond the ADOS

This section provides a discussion of potential generalization of our methods beyond the ADOS tool. We start by presenting our contribution of new data-driven descriptors through clustering, and end the section by showcasing potential application of the simulation method to other tools, such as personality assessment tests.

4.5.1 Generating new descriptors

The existing ADOS descriptors are useful for diagnosis and for informing decisions such as whether or not the child needs therapy. However, from a behavioral modeling point of view, these descriptors may be neglecting important behavioral aspects that are not directly related to a one-dimensional scale of autism severity. More specifically, there are two limitations to the existing descriptors for the purpose of simulation, as we explain next.

First, two subjects can have the same overall totals but very different subtotals. For instance, one subject might have a very high RRB subtotal and very low Comm. and Soc. subtotals and another might have medium values on all subtotals. In this case, it is not clear whether or not it is natural to group them under the same class. Second, although only 14 out of the 29 features have been identified as having enough predictive power when it comes to the autism severity, the remaining 15 features may carry useful information for the purpose of behavioral modeling. Also, the calculation of totals involves remapping, which reduces the resolution of some features by lumping values of 2 and above into one category.

In order to address these limitations, we use a data-driven approach to generate new descriptors obtained through *clustering* of the data points. Even though we established in Section 4.2.2 the absence of clearly separated clusters in the data, clustering algorithms effectively define regions of the feature space using the distribution of the data across that space. We refer to data points fallling in these regions as classes rather than clusters given the distribution of the data. To address the first limitation, we perform clustering in the 3D ADOS subtotal space to generate descriptor $D^{\text{data,lo}}$. To address the second, we perform clustering in the full feature space to generate descriptor $D^{\text{data,hi}}$. These descriptors take the form of a class centroid that indicates which class a given feature vector belongs to.

There exist many types of clustering algorithms, broadly categorized as density-based, distribution-based, connectivity-based, and centroid-based methods. Density-based clustering [56] assumes large density differences within and between classes, which from our SOM analysis is not a reasonable assumption. Distribution-based clustering [63] (e.g., using Expectation-Maximization over a Gaussian Mixture Model) assumes we know the distribution of the data, which is not a practical assumption since such domain knowledge is hard to approximate. Connectivity-based clustering [49] is not robust to noise and outliers, which makes it not suited for our noisy dataset. Therefore, we perform clustering on the data using a simple K-means [83], which is a centroid-based approach. The tendency of the algorithm to partition the data into equally-sized regions makes it desirable for our purposes. We select as our number of classes $K = 4$ (similar to the four severity classes from Figure 4.5). We use L1 distance as our distance function since we are dealing with discrete features. Figure 4.10 shows the clustering results as well as a visualization of how the obtained descriptors partition the data generated with DB-MMS. An analysis of the resulting class centroids is presented below.

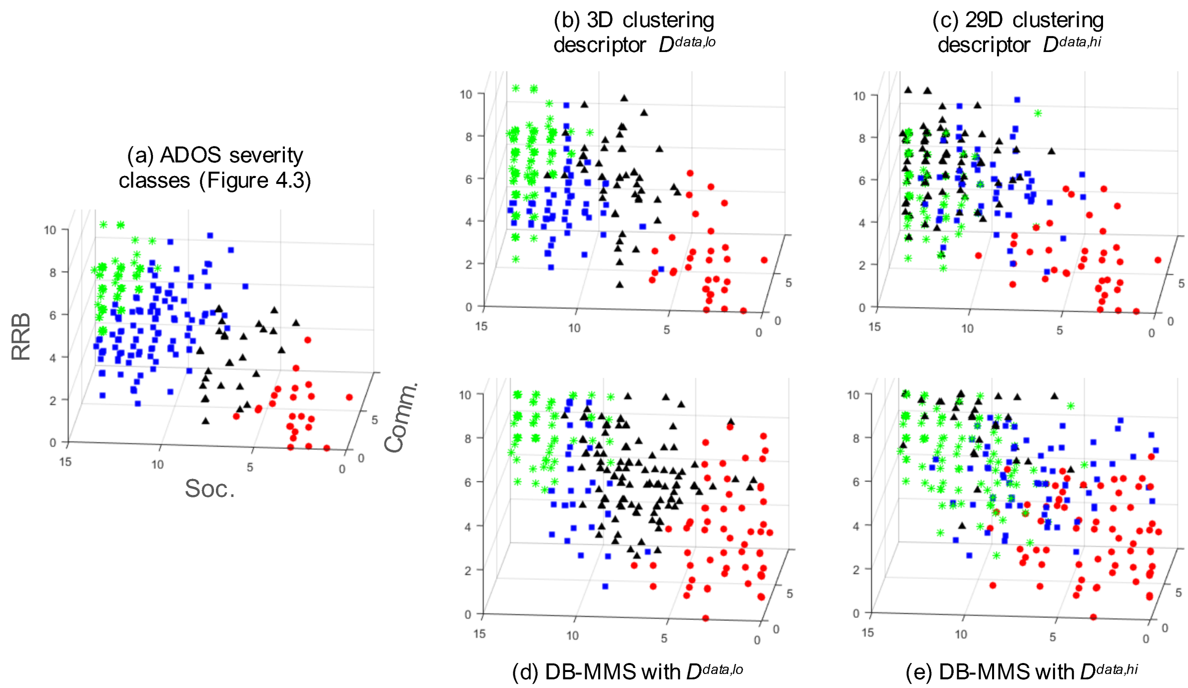

Figure 4.10 Generation of new data-driven descriptors through K-means clustering visualized in the ADOS subtotal space. Except for (b) and (d), as well as (c) and (e), points with the same color coding across plots do not belong to the same class, but were coded identically due to their similar location in the visualization space. *K-means parameters*: distance type: L1; max iterations per trial: 10,000; # trials: 10,000; initialization method: K-means++.

Low dimensional data-driven descriptor ($D^{\text{data,lo}}$)

The values of the obtained centroids are $\langle 1, 3, 1.5\rangle$, $\langle 3, 8, 5\rangle$, $\langle 5, 11, 3\rangle$, and $\langle 6, 13, 5\rangle$ (vector order is Comm., Soc., RRB). Figure 4.10(b) shows the real data points grouped according to this descriptor, and (d) shows a similarly sized synthetic dataset generated by DB-MMS using $D^{\text{data,lo}}$ as a descriptor. Comparing this partition to the severity partition (Figure 4.3) shows that $D^{\text{data,lo}}$ captures differences in the RRB subtotal not reflected in the 'Moderate' class of the severity partition.

High-dimensional data-driven descriptor ($D^{\text{data,hi}}$)

Similarly to $D^{\text{data,lo}}$, the resulting classes are visualized both on real data and synthetic data from DB-MMS in Figure 4.10(c) and (e). We analyze the resulting class centroids[4] by looking at the sample variance as well as the sample correlation across feature values for different centroids. The highest variance, corresponding to features that vary most across the four class centroids, occurs for feature 'Pointing' (PNT) followed by feature 'Overall Level of Non-echoed Language' (LANG). The lowest variance occurs for features 'Anxiety' (ANX) and 'Self-Injurious Behaviors' (INJ) where all four values are 0 for both features. The most negative correlation between pairs of features is -0.43, and occurs between 'Overactivity' (OVR) and 'Intonation of Vocalizations or Verbalizations' (IN)), and between OVR and 'Stereotyped/Idiosyncratic Use of Words or Phrases' (STER). However, since both IN and STER had a particularly significant number of `NaN` values replaced by random values, this last result might be due to noise. On the other hand, many features had a correlation of 1 across class centroids, indicating that it is more common to have similar trends in different feature values across classes as opposed to inversely related trends.

Figure 4.10(c) shows the real data points rendered in the ADOS subtotal space for easy visualization and grouped according to $D^{\text{data,hi}}$. Even though this descriptor still somehow encodes severity, it also captures specific differences that seem to vary more intensely across subjects such as pointing behaviors and use of language. Some overlapping points in the 3D space are even mapped to different classes, validating the fact that the ADOS descriptors neglect important features for behavioral modeling.

To sum up, we have illustrated how different types of descriptors can be used for simulation purposes. The data-driven descriptors provide additional insight about the distribution of the data in the feature space. We expect them to also be of value to the

[4] The full centroids for $D^{\text{data,hi}}$ are:
<1,1,0,0,0,0,0,1,0,0,1,1,0,0,1,1,0,1,0,1,1,2,0,0,0,0,1,0,0>
<1,1,2,1,2,0,1,1,2,1,1,1,1,0,1,1,2,2,0,1,1,1,1,1,0,1,0,0,0>
<3,2,2,1,2,1,3,2,2,2,2,2,1,1,1,1,2,2,0,2,2,3,2,2,0,2,1,0,0>
<3,2,1,2,1,2,3,2,2,2,2,2,2,2,2,2,2,2,2,2,2,3,1,1,0,1,2,1,0>

autism research community, as a preliminary unsupervised learning approach to identify the co-occurrence of behavioral patterns in individuals with ASD.

4.5.2 Applicability to other tools

The contributed methods presented in this chapter were formulated in a way that does not necessarily restrict them to the ADOS. In particular, DB-MMS applies to any instance where we have access to pairwise correlations between features, as well as a logical constraint of the form $\psi(\mathbf{f}, \mathbf{D})$. Similary, GBS was formulated in a way that applied to any case where one has a way of determining compatiblity between pairs of behaviors, either manually or using behavioral channels and compatibility rules.

We expect our contributions to be applicable with minor modifications to other interactive tools such as tools used in Applied Behavior Analysis (ABA) [44]. Examples include the Verbal Behavior Milestones Assessment and Placement Program (VB-MAPP) [144], the Assessment of Basic Language and Learning Skills (ABLLS) [64], or the Assessment of Functional Living Skills (AFLSTM) [116]. Other examples, not necessarily tied to ABA, but relevant to both therapy and education, include the Bayley-III [21], the Neonatal Behavioral Assessment Scale (NBAS) [28], Developmental Indicators for the Assessment of Learning-III (DIAL-III) [106], the Denver II [61], and the Parent-Child Interaction Assessment-II (PCIA) [19].

Beyond interactive tools, we expect a similar approach to be generally useful for robots expected to display consistent behavioral patterns throughout an interaction, such as for example personality traits or context-dependent attitudes. In particular, we believe these methods are partially applicable to tools for personality assessment used in psychology. One example is the Myers-Briggs type indicator (MBTI) [110] that uses five dimensions of personality assessment. Another example is the Minnesota Multiphasic Personality Inventory (MMPI) [69], a tool used to assess both personality traits and psychopathology (e.g., depression, hysteria, hypomania, social introversion, etc.). Although these tests are based on questionnaires hence not interactive by nature, they could still be useful to inform the behavior of a robot programmed to display certain human-like behavioral traits captured by these tools. More research is needed to assess the exact method by which questionnaire items can be translated into context-dependent robot behaviors.

All of these tools, similar to the ADOS, rely on scales that aggregate the receiver's responses on individual items into a profile of varying complexity. Because this mapping is algorithmic (and often computerized), it is generally possible to reverse the aggregation function the same way that was done with the ADOS. Furthermore, the correlations between scales and subscales are typically included as part of the tool manuals and can be used for generating synthetic realistic responses on individual items using the DB-MMS algorithm.

4.6 Related work

Simulating and modeling human behaviors is a widespread practice to inform any type of decision-making involving humans. Examples include consumer modeling in market research [7, 37, 139], online recommendation systems [98], and simulating vehicle driver behavior [32], to name a few. Before concluding the chapter, we discuss some work specifically related to user simulation for interaction research, as well as ASD-specific simulations.

4.6.1 User simulation for interaction research

User simulation is a common practice in computing-related fields to test the performance of algorithms, architectures, pipelines, or overall systems, mostly in the digital realm. With embodied and interactive technological artifacts, such as robots, operating in the physical world, Steinfeld et al. (2009) make the case for the need of simulating the human for the purpose of interaction research [142]. They argue that if the common 'Wizard of Oz' ('human-puppeteered' robot) approach is accepted in the field of HRI, then so should the inverse approach of simulating the human, to focus on technological advances. They call it the 'Oz of Wizard' approach and outline a number of conditions where such an approach could be justified. They also present a spectrum of combinations of 'Oz' and 'Wizard' that categorizes existing efforts in the field, and can help frame future ones. One important note in their work is the justification of "placeholder simulation using simplified human models", in contrast to "methodologically rigorous human modeling". The types of models used to enable such simulation can range from moderately precise, such as the work of Trafton et al. (2006), in which a child's thought processes during a game can be simulated [150], to highly simplistic, such as generating random inputs according to a Gaussian distribution centered around an estimated mean.

While these efforts have looked at simulating the user for the purpose of developing better technology to serve that user, our contributions in this chapter and the next look at a different role of simulation. Instead we use simulation for the ultimate purpose of serving a human-human interaction, namely complementing therapist training in preparation for encounters with real children. The simulation serves as a placeholder for the real human with which the therapists will interact in the future.

4.6.2 Simulation of ASD behaviors

Existing computational models of ASD include techniques such as neural networks or game theory to model low-level mechanisms of the brain affecting behavior [65]. These methods are able to explain different observed autistic behaviors, but not as successful in computationally predicting high-level behavior, especially for different types or severities of ASD. Reinforcement learning methods have been proven useful in modeling some high-level behaviors seen in individuals with ASD [22], but they are only able to distinguish between ASD and non-ASD populations. Moreover, a general purpose 'computer based mental simulator' (NL_MAMS) has been developed and used to simulate the underlying mental processes of individuals with ASD [65]. Finally, individual differences, well established in available diagnostic tools, are starting to be studied from a modeling/simulation perspective [136] but the parts of the model accounting for these differences is usually simplistic.

In relation to emulation of common ASD behaviors by robots, some work has been done on real-time motion imitation of children with ASD [146]. Additionally, some research looked at using robots as a platform to test theories related to low-level cognitive and sensorimotor processes related to ASD. They specifically look at aspects of behavior such as joint attention [130], or sensory integration and movement [18].

The literature discussed above is limited in that it does not provide us with the tools to develop an agent capable of interactions in a relevant clinical settings such as therapy or diagnosis. Our contribution in simulation of individuals with ASD was to leverage the intrinsic model of a diagnostic tool to simulate structured interactions of a spectrum of individuals without having to model low-level cognitive processes for which we do not have useful models.

4.7 Summary

In this chapter we discussed our approach for simulating behaviors of children with different severities of ASD, in a range of standardized tasks from the ADOS tool. While the ADOS maps child behaviors to an ASD severity value, our method aims at mapping a severity value (along with the age and language ability of the child) to a set of behaviors consistent with these descriptors. We contributed two algorithms, DB-MMS to sample feature vectors informed by ADOS data from 279 individuals, and GBS to select behaviors consistent with feature vectors while ensuring pairwise compatibility of selected behaviors. The end-to-end simulator, ADOS-Sim, integrates the two contributed algorithm as well as other components described in this chapter, and outputs behaviors as text. In the next chapter, we discuss our efforts towards embodying this virtual simulator of a receiver's behavioral responses into a robot capable of multi-modal interaction with the a provider (namely an autism therapist).

Chapter 5

Interactive Robots for Provider Training

This chapter builds and expands on the approach of the previous chapter by presenting an embodied and interactive platform for the simulation of receiver behaviors. Specifically, we enable the NAO robot to respond interactively in a restricted set of standardized ADOS tasks led by a therapist. We start by designing 16 'autism-like' robotic behaviors capturing different ASD severities along selected ADOS features. We then integrate these behaviors into an autonomous control architecture, allowing therapists to continuously interact with the robot through standardized ADOS-based tasks. Through individually controllable features, the robot can be customized in one of 256 unique behavioral profiles. We evaluate the validity of our interactive robot in both video-based and 'in situ' studies with three ADOS-certified therapists [13, 17]. We also present preliminary subjective evaluations on its potential benefits, including complementing existing therapist training.

5.1 Why robots with 'autism-like' behaviors?

Current therapist training for ASD diagnostic tools[1] heavily relies on videos and theoretical material, as well as observing a real diagnosis session run by a trained expert. Even though it exposes the therapists in training to a wide range of examples of behaviors and stresses on the rigorousness of the feature coding schemes and task procedures, it largely ignores

[1]https://www.wpspublish.com/store/c/343

the interactive and embodied component required for a successful administration of the tool. In fact, the interactive component represents a crucial part of the administration process. Given that therapists are expected to follow very specific sets of standardized instructions, while paying attention to behaviors, taking notes, and possibly adapting the order of tasks in real-time, a poor mastering of these interactive skills may result in mistakes in task administration as well as feature coding. A lowered reliability, especially in the coding of some features with already low agreement scores [101] defeats the purpose of using a standardized tool in the first place. Therefore, we propose that utilizing robots capable of exhibiting ‘autism-like’ behaviors may help complement existing training methods. These robots would equip therapists in training with a restricted but powerful interactive simulation environment to train on safely before moving on to scenarios involving real children.

A robot capable of simulating ‘autism-like’ behavioral responses has the following advantages:

- *Interactivity* — Unlike existing therapist training methods, a robot is capable of simulating, to a limited extent, the structured interactions of an ADOS session.

- *Customizability* — In the real world, the experience therapists gather depends on the patients they receive, which is difficult to control. The customizable aspect of our robot allows to generate arbitrary behavioral profiles, greatly increasing the number and diversity of feature combinations the therapists can be exposed to.

- *Repeatability* — Real-life interactions happen only once, and if we attempt to repeat them, there will always be some inevitable differences. Even though videos showing behaviors or interactions may be repeated to be better studied, the use of an interactive robot allows the interaction itself to be repeated in a controlled way, and allows for reiterating previous interactions in the event of procedural errors, or lack of observational attention.

Furthermore, research on human perception has shown that people tend to assign human-like traits to technological artifacts, including robots, perceiving them as social beings [54]. This aspect of our cognition motivates the use of humanoid robots that do not necessarily have to reproduce the physical appearance or size of a child with high fidelity. In fact, the NAO robot is much smaller than a young child, but possesses basic features that make it expressive and able to exhibit engaging social behaviors.

5.2 Interaction design

The interaction between the robot and the therapist is structured into independent ADOS-based tasks, each of which has an associated action scale that the therapist can use to prompt the robot. On the robot side, we selected four features to characterize the robot's responses in these tasks. We designed 16 behaviors on the robot and integrated them into an autonomous control architecture that can be customized according to a user-controlled feature vector. The robot is able to automatically detect interaction parameters such as verbal and non-verbal actions as well as sound location, allowing for more natural and flexible interactions. In this section, we provide some details on each interaction component.

5.2.1 Tasks

In our interaction setup, the robot is standing on a table, at which the human is seated. The human also has objects available for each task if needed. In the video-based study we used arbitrary objects, while in the 'in situ' study with therapists we used objects that would be used in an actual ADOS session. Of the three ADOS-based tasks considerd in this chapter, the first two are similar to those considered in Chapter 2, with human and robot roles reversed, and the third one is new. They are listed below:

- **'Name Calling' task (NAME)** — The human performing the actions direct the attention of the robot to themselves. For this task, we assume that two humans are present: a 'familiar' person and the therapist (considered 'non-familiar').
- **'Joint Attention' task (JATT)** — The human directs the attention of the robot to an object on the table. The object can be activated to produce sound if needed.
- **'Snack Preference' task (SNACK)** — The human asks the robot to express a preference between two snack options placed on the table.

The tasks are independent and do not have to occur in any particular order.

5.2.2 Action scales

In Chapter 2, action scales were defined for the robot to execute. In this chapter, these actions will be performed by the human. For each task, we clearly define the actions that the robot would expect to perceive from the human. In particular, we define the exact content of the speech, as well as any non-verbal behaviors that the robot can perceive, as summarized in Table 5.1. Figure 5.1 further shows snapshots of humans performing one of the action scales.

The robot expects the human to go through these actions hierarchically from lowest to highest level, but allows for an arbitrary number of repetitions of individual actions.

Table 5.1 Summary of defined action scales the human can use to interact with the robot.

Task	Goal behavior	Level	Action (performed by human)
NAME	Robot looks at human	1	"NAO!"
		2	Ask 'familiar' person to say "NAO!"
		3	Touch robot's head
JATT	Robot looks at target object	1	"Look!"
		2	"Look at THAT!"
		3	Activate object
SNACK	Robot expresses preference	1	"Which snack do you like?"

Figure 5.1 Example of an action scale performed by two researchers (task NAME).

5.2.3 Robot behaviors

We focused on four features from the ADOS Module 2 to inform our design of robotic behaviors that emulate those of children with varying ASD severities. We used Module 2 instead of 1 because it contained richer sets of behaviors to showcase the feasibility of our research goals. However, the same method described in this chapter can be applied to any of the ADOS modules. The features considered in this chapter are: 'Response to Name' (RNA), 'Response to Joint Attention' (RJA), 'Overall Level of Non-echoed Language' (LANG), and 'Pointing' (PNT). As is the case with a child, those features can characterize the behavioral responses of our robot to task actions. Each of the selected features takes on discrete values between 0 and 3, where higher values correspond to higher levels of impairment or deviation from the typical response, as was described in previous chapters.

Based on the behavior database discussed in Chapter 4, we designed one representative behavior for every feature value, resulting in a total of 16 robot behaviors summarized in Table 5.2. We chose the representative behavior according to how translatable it was to the NAO platform. A robot behavior consists of an animation of the robot's joints as well as possibly speech. It is triggered by one or more task actions, depending on the behavior. Unlike Chapter 2 where feature values only characterized which actions cause a goal behavior, in this chapter we also consider the quality of the behavior, informed by ADOS specifications. Furthermore, some of our behaviors are parametrized (e.g., gaze behavior takes as a parameter a 3D location to look at). In the presence of more than one relevant feature for a task (e.g., SNACK), behaviors are blended, meaning they are run simultaneously.

5.2.4 Autonomous control architecture

We integrated our designed behaviors as part of an autonomous control architecture, enabling continuous interactions with one or more humans, according to the actions the robot recognizes. More importantly, the robot can be customized by specifying an arbitrary value for each feature, resulting in 256 unique customizations.

Figure 5.2 summarizes the autonomous robot control architecture. A perception module recognizes and discriminates between the different actions performed by the human, triggering an appropriate response on the robot. The behavior selector uses the values on the task-relevant features to select appropriate behaviors from the database. Finally, the behavior generator instantiates the behaviors with the appropriate parameters (e.g., sound location) and blends them if more than one behavior was selected (only for

Table 5.2 Summary of the designed ‘autism-like’ robot behaviors of varying severities.

Task	**Relevant feature(s)**	**Behaviors** Value 0	Value 1	Value 2	Value 3
NAME	RNA	Looks at human within second name calling attempt with coordinated utterance “Yes?” $b(\mathrm{RNA}, 0)$	Same as $b(\mathrm{RNA}, 0)$ but only responds to ‘familiar’ human while ignoring ‘non-familiar’ one $b(\mathrm{RNA}, 1)$	Looks in general direction of ‘familiar’ human only (without eye contact or utterances) while ignoring ‘non-familiar’ one $b(\mathrm{RNA}, 2)$	Only responds to touch on head by exhibiting succession of random gaze shifts; ignores all other actions $b(\mathrm{RNA}, 3)$
JATT	RJA	Immediately looks at object, then human, then back at object $b(\mathrm{RJA}, 0)$	Ignores first action level; looks at object only at second action level “Look at THAT!” $b(\mathrm{RJA}, 1)$	Ignores first two action levels; only looks at object when activated and emitting sound $b(\mathrm{RJA}, 2)$	Same as $b(\mathrm{RJA}, 2)$ but with slight gaze shift towards object without actually looking at object $b(\mathrm{RJA}, 3)$
SNACK	LANG	Says: “I like this snack of all the snacks in the world.” $b(\mathrm{LANG}, 0)$	Says: “This one.” $b(\mathrm{LANG}, 1)$	Says: “This.” $b(\mathrm{LANG}, 2)$	Echoes: “Snack... Snack... Snack... Like... Like...” $b(\mathrm{LANG}, 3)$
	PNT	Clearly points at one of the snacks with coordinated eye gaze $b(\mathrm{PNT}, 0)$	Clearly points at one of the snacks with slight gaze shift not in direction of pointing $b(\mathrm{PNT}, 1)$	Looks at one of the snack but without pointing $b(\mathrm{PNT}, 2)$	Slightly shifts gaze downwards with no pointing $b(\mathrm{PNT}, 3)$

task SNACK in our implementation). The value of individual features is allowed to be controlled by a user. Depending on the aim of the interaction, and whether the therapist should be agnostic or not to the feature values, this user could be either the therapist or another person. We implemented this architecture on the NAO robot using the NAOqi Python API through the Choregraphe suite[2].

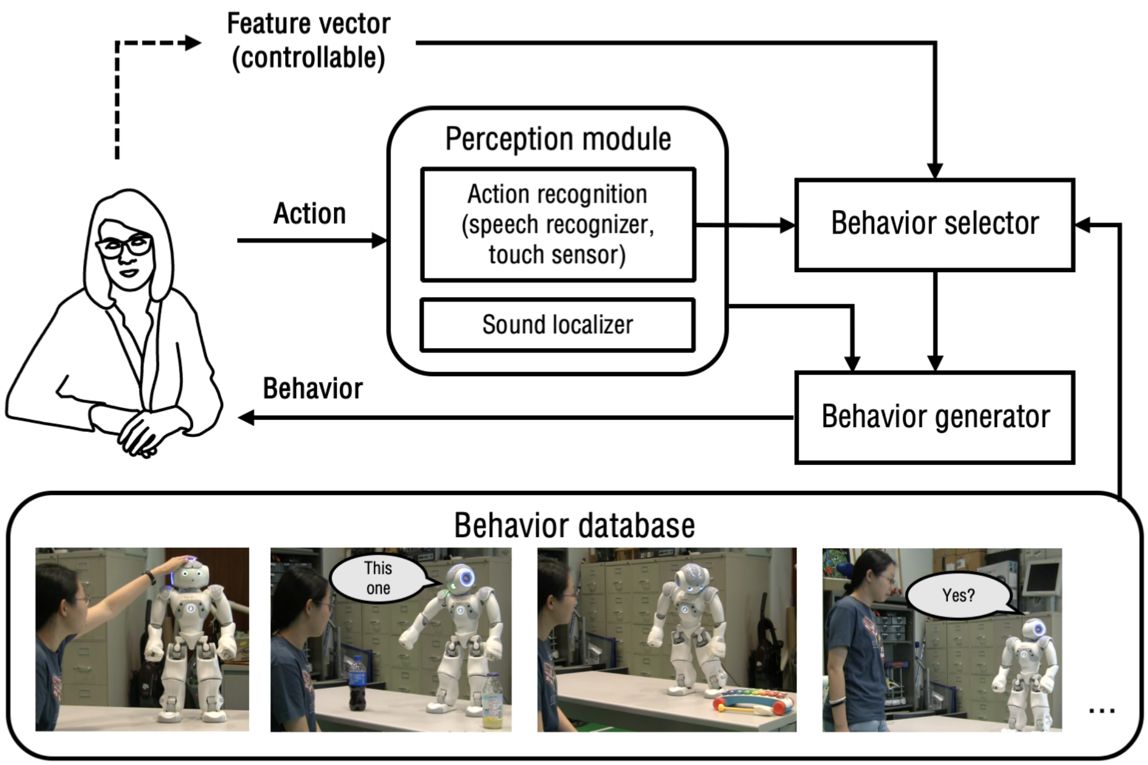

Figure 5.2 Overview of the autonomous robot architecture.

Because of the robot's perceptual limitations, some parameters needed to be hardcoded or are estimated simplistically, while others are easier to detect completely autonomously. Below are some more details on the parameters automatically estimated versus hardcoded:

- $b(\mathrm{RNA}, 0)$ *through* $b(\mathrm{RNA}, 3)$ — The voice location is estimated using NAO's microphone array and used to modulate the robot eye gaze. The 'familiar' and 'non-familiar' humans are distinguished simplistically, based on the location of the voice. We assume that the 'familiar' person would always be on one side of the

[2]Code available at https://github.com/kobotics/autistic_nao

robot (e.g., left) and the 'non-familiar' always on the other (e.g., right). The touch sensor on NAO's head is used to trigger $b(\mathrm{RNA}, 3)$.

- $b(\mathrm{RJA}, 0)$ *through* $b(\mathrm{RJA}, 3)$ — Because of the robot's perceptual limitations, the location of the object used for JATT is hardcoded in $b(\mathrm{RJA}, 0)$ and $b(\mathrm{RJA}, 1)$. For $b(\mathrm{RJA}, 2)$ and $b(\mathrm{RJA}, 3)$, it is estimated using sound localization, since object activation emits a sound. For motion stability purposes (robot loosing balance at times), the location of the human in the JATT is also hardcoded.

- $b(\mathrm{PNT}, 0)$ *through* $b(\mathrm{PNT}, 3)$ — The positions of the two snacks on the table are hardcoded. The preferred snack position is used to parametrize the eye gaze and pointing directions of the robot.

- $b(\mathrm{LANG}, 0)$ *through* $b(\mathrm{LANG}, 3)$ — These behaviors consist of speech only, and are not parametrized.

For all behaviors, the speech recognizer is used to detect verbal actions, which triggers the corresponding responses, when applicable. When idle, the robot is animated through a subtle 'Breathing' behavior in which the robot slightly shifts its weight from one foot to the other. A video showing sample human interactions with our autonomous NAO robot in 'low ASD severity' and 'high ASD severity' modes is available for online viewing[3].

5.2.5 Behavior evaluation (video-based study)

In order to evaluate the validity of our designed interactive behaviors with respect to the formalism of the ADOS, we ran a first video-based study with ADOS-certified therapists. The aim of the study was to investigate: (1) whether the therapists would assign to the features characterizing the designed behaviors the same values as the ones on which their design was based, and (2) whether the therapists would agree with each other in their evaluation, and how this agreement would differ across the different robot behaviors.

Survey structure

The study consisted of a video-based survey showing short videos[4] of the isolated designed behaviors in the context of an interaction with a human (or two for the behaviors requiring

[3] Video available at https://bit.ly/2tE2nOD
[4] Survey videos available at https://bit.ly/2MqRdDK

more than one person). In this study, the people in the videos were researchers that contributed to the development of the robot behaviors. Based on what they saw in the video, the participants provided a value between 0 and 3 on the relevant feature(s) of each video, according to the description for each feature value in the ADOS manual. Detailed instructions were given in relation to feature coding, background on robot's capabilities, and simplifying assumptions. In particular, the participants were instructed to 'diagnose' the robot the same way they usually do it with children, by coding the feature value they thought best characterized the response they observed in the video. They had the possibility to watch the video as many times as needed. Also, they were instructed to use information from the current video only and after the first action was started (even though some of the features usually require several samples to form a good judgment). Finally, they were asked to ignore any expression unrelated to motion or speech, including non-verbal cues acknowledging the detection of speech, namely beeps and color changes of the NAO's eyes. These cues, part of the default behavior of the speech recognizer, were kept in our interaction because they were designed to facilitate speech synchronization and the debugging of the state of the robot in case of a recognition failure.

The videos were grouped according to the three tasks (NAME, JATT, and SNACK). Because behaviors were blended in task SNACK, and to avoid overwhelming the participants with a very large number of videos, we chose to set the feature values to be identical, in all videos for that task, for both language and pointing features (i.e., $\langle b(\mathrm{LANG}, 0), b(\mathrm{PNT}, 0)\rangle$; $\langle b(\mathrm{LANG}, 1), b(\mathrm{PNT}, 1)\rangle$; ...). We had a total of 12 videos, four for each of the three tasks. To avoid any order effects, we randomized the order in which the tasks were shown as well as the order of the videos within each task. In cases when the robot was intentionally programmed to ignore certain actions, the human(s) in the video performed the actions hierarchically until the robot responded. The survey also included relevant snapshots of the ADOS manual to refresh the memory of the trained experts and minimize errors in their coding.

Methodology

We first ran a small pilot with one ADOS-certified therapist to gather feedback on the clarity of the survey and the videos, and potential points for improvement. When the survey was finalized, we gathered the online responses of three other therapists from the Child Development Center at the Hospital Garcia de Orta in Almada, Portugal. The therapists who participated in this study were all women who received some form of ADOS training. Informed consent and permission to use media was obtained at the beginning of the survey. Results of the study are discussed in Section 5.3.

5.2.6 Interaction and potential benefits assessment ('in situ' study)

The aim of this second study was to test our robot under different configurations in a real interactive setting with autism therapists, as well as assessing the potential benefits of this interactive robotic tool. This study therefore relied on, first, coding the robot behaviors according to the ADOS specifications and, second, answering a questionnaire we devised to assess the potential benefits of our robot in real-world applications. This study was performed with the same three participants from the video-based study, 11 months later.

Methodology

In the main part of the study, the participants interacted with the robot through the set of tasks we defined, observing and subsequently coding the robot's responses according to the ADOS specifications, as was done in the video-based study. The robot configurations were similar as well (matching severities on language and pointing features), and we exposed the participants to the same 12 robot responses. However, there were some differences as compared to the video-based study:

- In the video-based study, we consecutively showed different robot responses for the same actions to allow for better comparison of behaviors, as the focus was solely on validating the behaviors themselves. In this study, we were interested in a more naturalistic and holistic evaluation of the interaction, going beyond isolated behaviors. As a result, we had the participants go through each task once, then repeat the process, with four different robot customizations randomly permuted while ensuring that each robot behavior appeared once. This way, participants could get a sense of an entire interaction with four 'different' robots that they would have to diagnose, similar to four different ADOS sessions.

- As participants were allowed to replay the videos in the previous study, in this study, they were allowed to repeat the task as many times as needed for coding the behaviors.

- Within the constraints imposed by our robot, we tried to replicate as much as possible the physical setting that the therapists are used to. For example, we used objects from the ADOS kit, such as one of the activatable toys from the ADOS, as well as one savory and one sweet snack. These objects differed slightly from the ones used the videos.

In addition to coding behaviors, we also asked participants to provide answers to a questionnaire, separated into two parts. The aim of the questionnaire was to compare the ratings of existing training solutions with our proposed solution, as well as to evaluate the potential benefits of robots with 'autism-like' behaviors in our foreseen applications.

The questionnaire structure is summarized in Figure 5.3. The first part, presented before the interaction with the robot, started by gathering background information about the participant's diagnostic training. We then asked the participants to assess that training along three dimensions, namely:

- *Behavior accuracy*, i.e., to what extent behaviors encountered in training match those encountered in real sessions.

- *Interactivity*, i.e., to what extent it involves an interaction, be it in a real or virtual setting.

- *Diversity of behavioral profiles*, i.e., to what extent combinations of feature values encountered are diverse.

Finally, we asked how much they believed robots with 'autism-like' behaviors could benefit our foreseen applications, namely:

- Complementing existing ADOS *therapist training*.

- Enabling new types of scenarios for *autism therapy* (e.g., imitation tasks).

- *Educating and sensitizing* the general population about the behavioral differences in children with ASD (e.g., classrooms, museums, workplace, etc.).

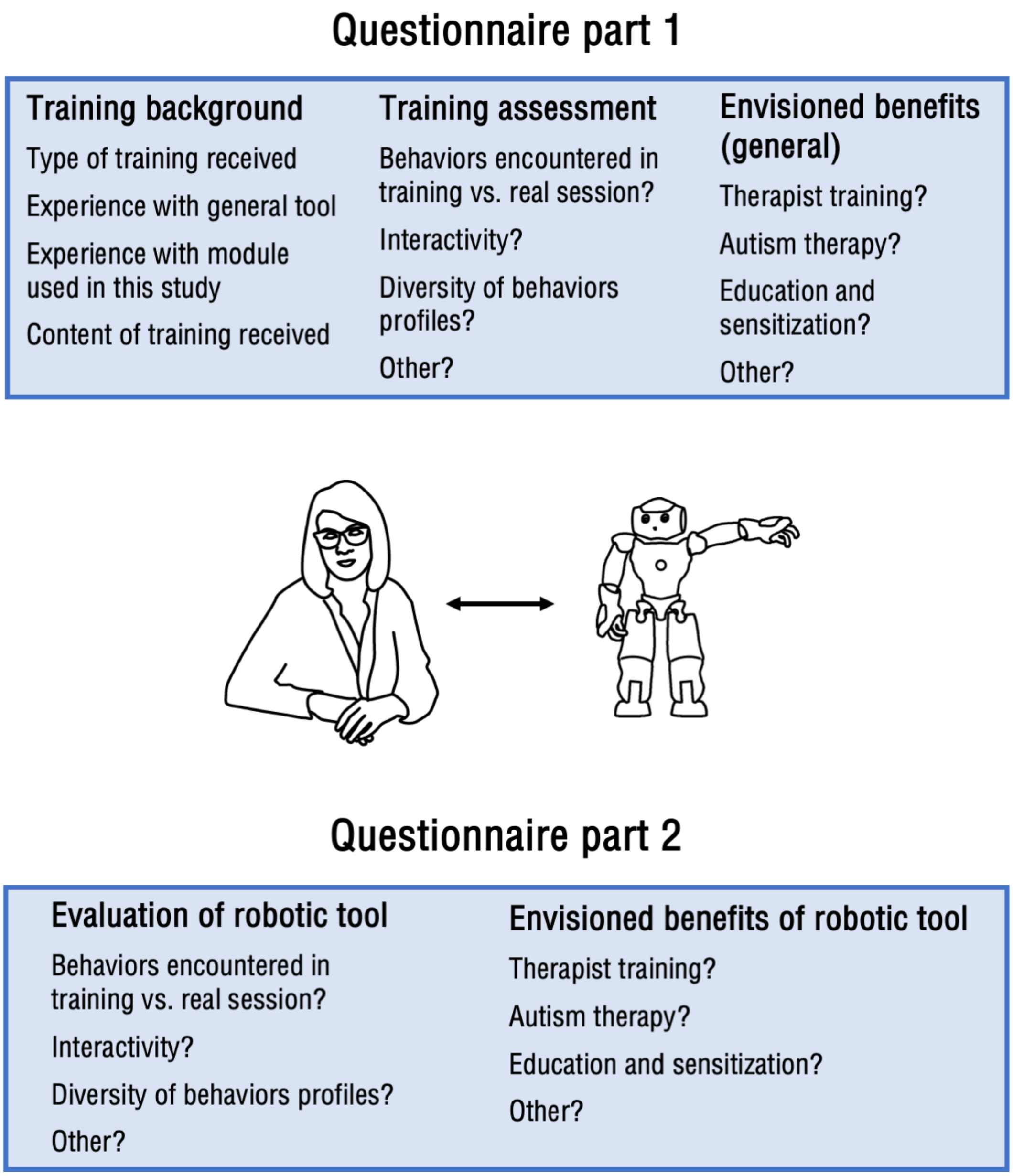

Figure 5.3 Questionnaire structure organized by categories and items within each category. Every item marked with a '?' is associated with a Likert item.

We further discuss the last two applications towards the end of this chapter in Section 5.4.

The second part of the questionnaire, presented after the interaction with the robot, repeated the same questions as the first part, but this time assessing specifically our robotic tool. Apart from the 'Training background' section, which was multiple choice, all responses were in the form of five-point Likert items. The questionnaire was in Portuguese, the participants' native language.

Procedure

After signing an informed consent, the participants filled the first part of the questionnaire. The examiner then took them into the robot experiment room and provided them with instructions on the tasks they were going to perform on the robot, as well as the structure of the rest of the study. Sheets with all needed information were made available to them, including the list of valid actions for each task, relevant snapshots from the ADOS, and space to use for coding. In addition to notes mentioned in the video-based study, the examiner also stressed that it was important that they spoke clearly and loudly, and that the robot only responded to voice and touch but not visual cues such as gaze or direction of pointing. As in the video-based survey, the examiner reminded the participants to only consider in their coding the robot's behavior after the first action of a given task was started. Participants were also asked to ignore any robot expressions unrelated to motion or speech.

Once any doubts they had were clarified, the participants 'diagnosed' the robot with the first customization going through the three tasks sequentially NAME – JATT – SNACK, observing the robot's responses and reporting their coded feature values on the sheet. Once the three tasks were over, the examiner announced that he was going to reprogram the robot, and asked the participant to treat it as a 'new robot'. The process was repeated until all four pre-randomized robot customizations were shown. Figure 5.4 shows some snapshots of these interactions.

Because of technical limitations of the robot, there were moments where the examiner had to briefly intervene, saying things like "the robot did not understand what you said, please repeat". The examiner, although present in case of doubt from the participant's part, tried to be as non-invasive as possible to maintain the naturalness of the interaction.

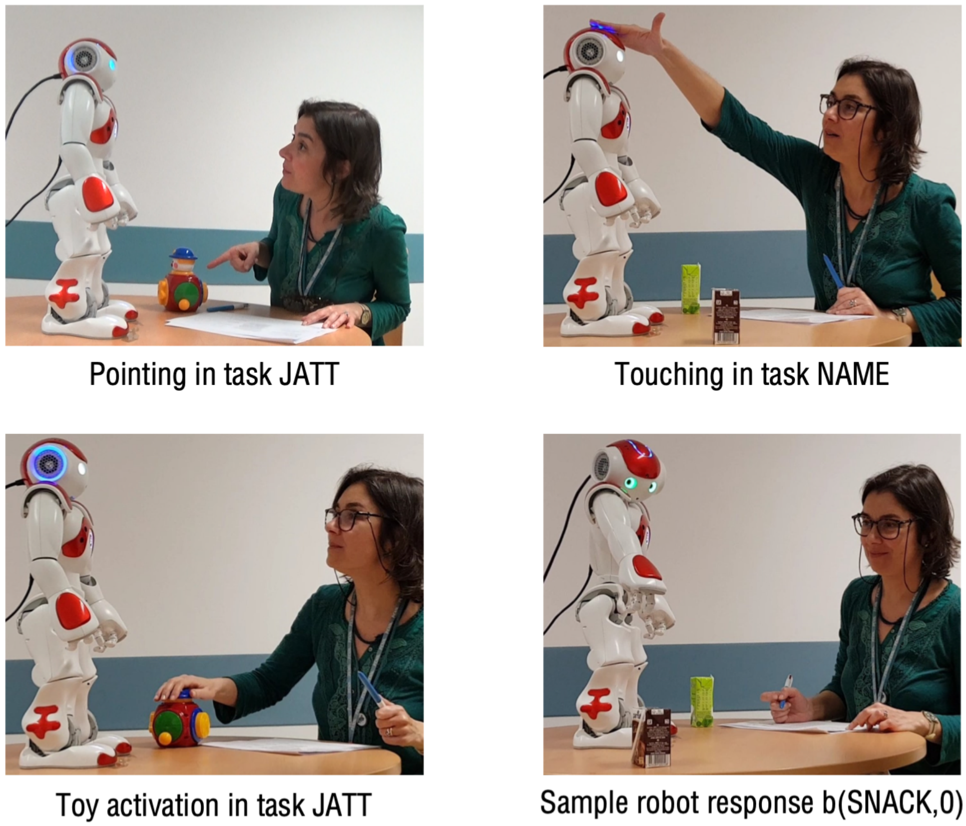

Figure 5.4 Autism therapist interacting with the robot and coding its responses according to the ADOS specifications. Images are shared under participant's informed consent.

5.3 Results and discussion

We first analyzed, across the two studies, the accuracy of responses and the agreement between the participants, summarized in Tables 5.3, 5.4, and 5.5. In our accuracy analysis, we only discriminated between correctly and incorrectly classified responses (with respect to the expected response). In the agreement analysis, we additionally treated the variables as ordinal in one of our metrics, and compared some of the results with reference values from real ADOS settings. We additionally investigated order effects in the responses. Finally, we analyzed the results of the questionnaire and compiled additional qualitative observations.

5.3.1 Accuracy results

The participants achieved an overall accuracy of 76.04% across the two studies. The accuracy was considerably higher for the video-based study (83.33%), as compared to the 'in situ' study (68.75%), with close to statistical significance using a McNemar's mid-p test on the overall binary categorical data ($p = 0.057$), and actual statistical significance only for expert 1 ($p = 0.031$). The same test showed no statistical significant difference in accuracy between all pairs of raters ($p \geq 0.125$ for all pairs), regardless of the type of interaction (video/real). There also seemed to be a relationship between the level of experience and the accuracy of the participants. The accuracy results per participant and per feature are summarized in Table 5.3 and Table 5.4, respectively.

Table 5.3 Accuracy results per participant across the two studies.

Expert	**Total accuracy (%)**			**ADOS training**	**Real-world experience**
	Video	Real	Both		
E1	95	63	78	Non-official	Low
E2	88	75	81	Official	High
E3	69	69	69	Official	Very Low

Table 5.4 Accuracy and agreement results per feature.

Feature	**Accuracy (%)**			**Agreement (r_S)**		
	Video	Real	Both	Video	Real	Both
RNA	75	75	75	0.91	0.85	0.80
RJA	100	83	92	1.00	0.76	0.86
LANG	92	75	82	0.97	0.91	0.92
PNT	67	42	54	0.93	0.59	0.76
Combined	**83**	**69**	**76**	**0.92**	**0.76**	**0.83**

Looking at individual features, RJA had the highest accuracy (91.67%), and PNT the lowest (54.17%). This result was expected as the latter was the most complex feature to code, and was paired with speech behaviors within the same task, possibly resulting in some interaction effects. However, the hypothesis that blending behaviors from supposedly independent features may involve interaction effects in the coding of individual features needs to be investigated more carefully with a larger sample. Moreover, we expected feature LANG to have the highest accuracy as it was the least subjective feature to code, which was not the case. We hypothesize that it may have also been subject to the interaction effect discussed above, as well as the fact that it was the only feature that differed considerably between Module 2 and Module 1 of the ADOS, the latter being the one that the participants were most used to in their professional practice.

To understand better the sources of misclassifications, we report confusion matrices for each feature in Figure 5.5. We can see that 11 out of 16 behaviors had an overall accuracy superior to 80%. Note that feature RJA showed relatively high accuracy for all behaviors. On the other hand, the two behaviors which had the lowest accuracies were $b(\mathrm{PNT}, 1)$ and $b(\mathrm{PNT}, 2)$. In some cases, it seemed that the participants thought it would be appropriate to code the gaze behavior as part of the pointing behavior, which should not have happened given that eye gaze is typically coded in a separate feature (not included in these studies). In other cases, on the contrary, participants seemed to have completely denied the importance of gaze for the pointing feature, which is justifiable. Behavior $b(\mathrm{PNT}, 1)$, containing a clear pointing, despite an uncoordinated eye gaze, was misclassified as $b(\mathrm{PNT}, 0)$ 83.33% of the time, which may suggest that this particular behavior would have to be redesigned and made clearer. For $b(\mathrm{PNT}, 2)$, the results were much more spread out, and we hypothesize that the source of misclassifications is a combination of low legibility of gaze behavior on the robot's part as well as lack of rigorousness on the participants' part.

The misclassifications for $b(\mathrm{RNA}, 0)$ came from the same expert, which may suggest that in this case the source of confusion was not from the robot, but from her relatively low experience level with the ADOS. The low accuracy of $b(\mathrm{RNA}, 2)$ is most probably due to the difficulty in assessing the gaze direction of the robot, as it seemed to be easily confused with $b(\mathrm{RNA}, 1)$, whose main difference is the direction and duration of gaze. For feature LANG, it seems like $b(\mathrm{LANG}, 3)$ (echolalia, which is easily identifiable) was the only behavior that was immune to misclassifications, while the other three behaviors seem to have been somehow affected by the factors discussed above.

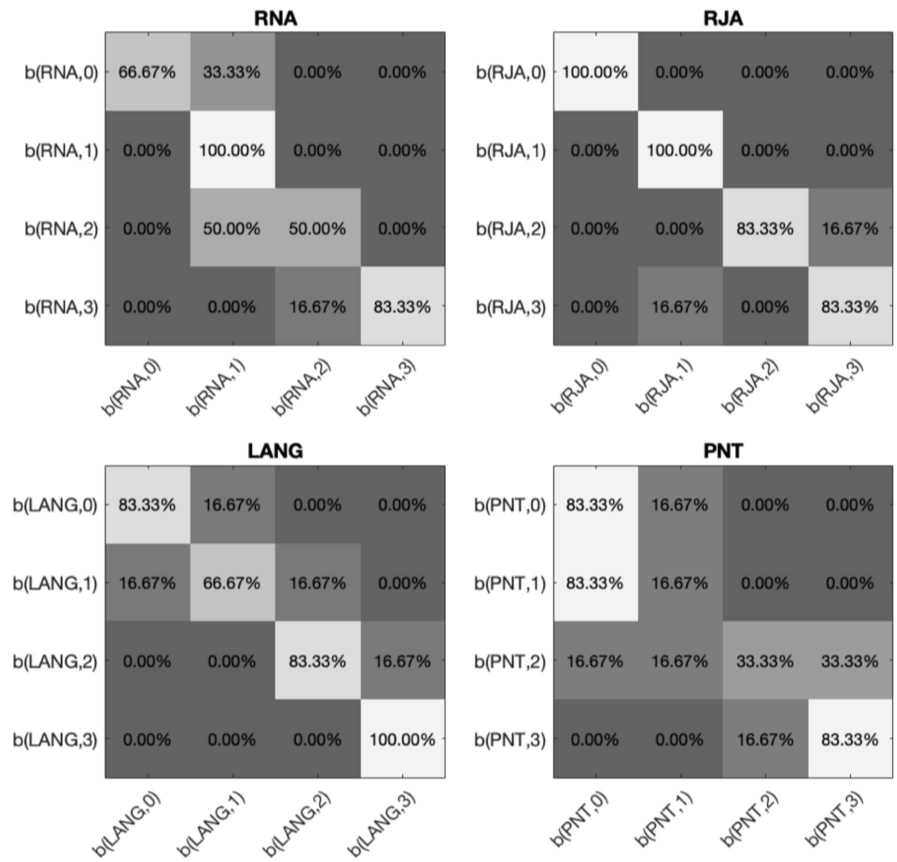

Figure 5.5 Confusion matrices for each feature across the two studies, relating the participant responses (x-axis) to the expected responses (y-axis). 11 out of 16 behaviors have an accuracy above 80%.

5.3.2 Agreement results

In our agreement analysis, we used two metrics. The first is the average Spearman's correlation coefficient (r_S) between all pairs of raters, which accounts for the ordinal nature of the data (last column of Table 5.4). The second is the percent agreement, which only discriminates between matching and non-matching responses, and was used for comparison with reference values from the literature on video-based ADOS coding [101, 154] (Table 5.5).

Starting with the average Spearman's correlation, we obtained a very high agreement value for both studies combined ($r_S = 0.83$). We computed p-values for each pair of

Table 5.5 Percent agreement across participant responses, in comparison with values from the ADOS literature with children. The naturalistic setting (Zander et al. (2016) [154]) reports values obtained from clinically trained ADOS users, while the ideal setting (Lord et al. (2012) [101]) reports values obtained from 'research-reliable' experts. All values relate to a *video-based coding* setting.

Feature	**Percent agreement (%)**		
	Robot	Children (naturalistic) [154]	Children (ideal) [101]
RNA	50	76	84
RJA	100	78	96
LANG	83	80	96
PNT	58	60	85
Combined	**73**	**74**	**90**

raters against the alternative hypothesis that the correlation is greater than zero, using the exact permutation distributions, yielding $p \leq 10^{-6}$ for all three pairs of raters, hence indicating general strong agreement between the experts, as expected. Similar to the accuracy results, agreement results differed considerably across the two studies ($r_S = 0.92$ for 'video' and $r_S = 0.76$ for 'real'). For both accuracy and agreement results, it is unclear if these differences were mainly due to the embodiment factor, or if the different grouping of behaviors played a role (blocks with the same task versus blocks with the same robot customization). Looking at individual features, the feature with the highest agreement was LANG ($r_S = 0.92$), which is expected given its highly objective coding scheme. The lowest agreement was for PNT ($r_S = 0.76$), which also showed a surprisingly large difference in agreement between the video and real scenarios. We attribute this difference to the same reasons that may have affected accuracy.

The percent agreement yielded lower or equal values as compared to the previous metric, as expected, since it considers that all mismatches have the same weight, achieving an overall value of 72.75%. In Table 5.4, we report for comparison the same metric values from two different sources of the ADOS literature with children. The last column reports values from the ADOS Module 2 manual by Lord et al. (2012), obtained from 'research-reliable' ADOS therapists under ideal conditions [101]. The middle column reports values obtained in a more naturalistic setting by Zander et al. (2016), from clinically trained

ADOS users pertaining to 13 different clinical sites [154]. Our robot behaviors achieve an agreement similar to the naturalistic setting case, while the ideal setting case shows much larger values. This result suggests that the sources of disagreement in our solution may be largely due to the common problem of rater subjectivity for non research-reliable ADOS users. This results supports the applicability of the ADOS tool to robotic behaviors.

5.3.3 Order effects

An additional hypothesis on misclassifications is that the participants may have gotten fatigued as the studies progressed. If this were the case, we would expect a positive correlation between the presence of errors and the index at which the behaviors appeared in the study, given the fact that counterbalancing was used. To test this hypothesis, we computed the Spearman's correlation coefficient between those two variables with a t-test for statistical significance. We found a statistically significant positive correlation ($r_S = 0.33^*$, $p = 0.022$) in the video-based study, which suggests that participants were getting fatigued as the survey progressed, making them prone to less sharp judgment. However, interestingly, this effect was not observed with the real robot ($r_S = 0.06$, $p = 0.336$), which we may attribute to the fact that the interaction was more engaging than answering an online survey. Also, the physical presence of the examiner and the learning effects may have contributed to that difference.

5.3.4 Questionnaire results

Figure 5.6 reports the results of the questionnaire responses. Since each item only had three responses, statistical tests will not be used in our analysis, however comparing the mean responses in each category may be indicative of expert opinion and is useful for directing future research endeavors in this space. We also show individual responses on each item for a better understanding of the data.

Overall, the participants provided high ratings for all three applications we suggested. It is interesting to see that, even though they had previously seen videos of our robot in the first study, their ratings on suitability for therapist training as well as therapy tasks increased after they had actually interacted with our robot.

In the particular application of therapist training, on average, our solution was rated higher than existing therapist training methods along the dimensions investigated. As expected, our solution was rated as much more interactive than existing solutions. It was also rated as similar in terms of profile diversity and lower in terms of behavior diversity, which is expected since our current prototype only considered three tasks and a single behavior for each feature value. In a future expanded version of this implementation, several behaviors could be considered for the same feature value.

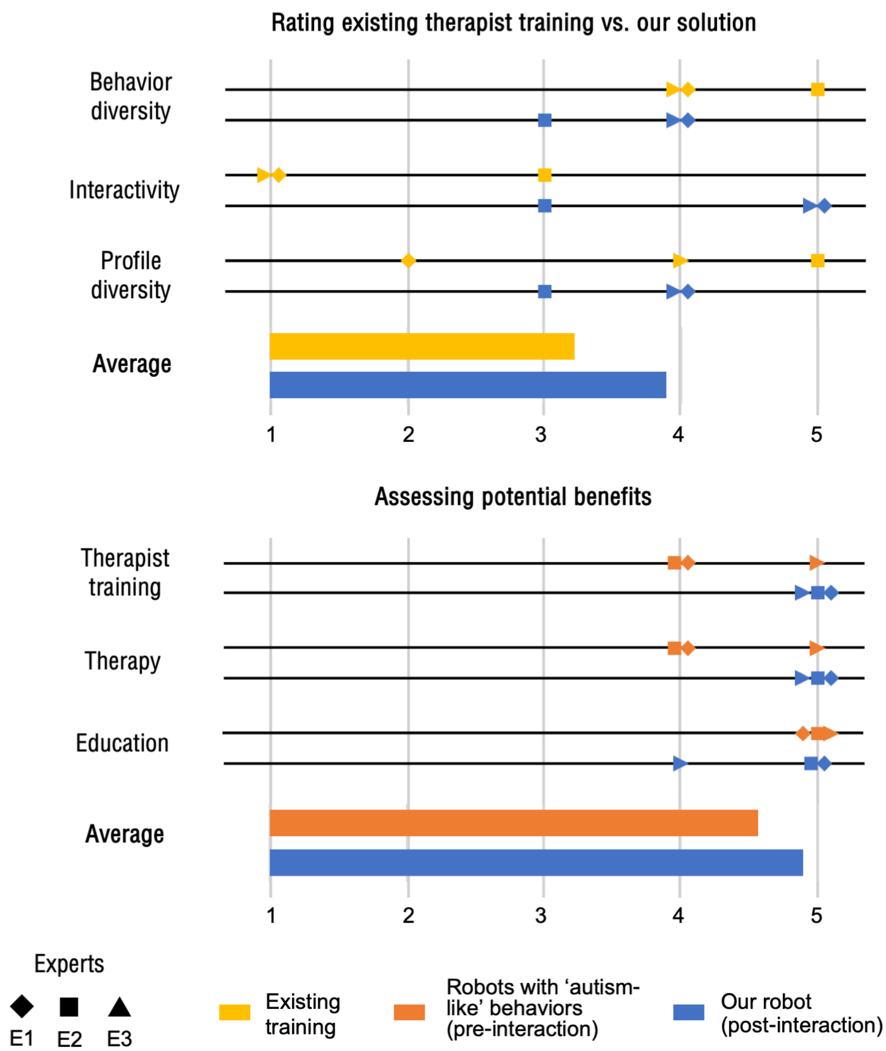

Figure 5.6 Summary of responses to questionnaire items. Overlapping points were slightly disturbed for better visibility.

5.3.5 Qualitative observations

Even though at first the participants expressed some skepticism towards our idea of 'simulating children with ASD', after both studies they seemed to be pleasantly surprised by how useful this robotic tool could be. They specifically emphasized characteristics of the robot that they had not foreseen. For example, they understood better that what we were trying to replicate was not the underlying cognitive mechanisms of children with ASD, but rather high-level behaviors that are clearly laid out and categorized in available diagnostic tools.

After the video-based survey, expert 2 said she was surprised that the interactions shown in the videos felt "just like ADOS tasks with real kids". After the 'in situ' study, she stressed that what she found very interesting and useful was that she could repeat the same action and observe the same response as many times as she wanted, which she thought made the system particularly suited for therapists in training.

Expert 1 mentioned that she was having trouble keeping up with the interaction, especially when it came to assessing the gaze direction of the robot. This difficulty may explain that she had the lowest accuracy in the 'in situ' study, but is unclear if it was because of the robot or because her level of experience was low (she had actually never performed a real entire session with a child). This observation motivates the importance of the interactive component needed for training, which our solution attempts to address.

5.3.6 Additional remarks

Perhaps the strongest limitation of our methodology was the small number of participants, as well as the lack of ADOS research-certified (as opposed to clinically-certified) participants, which we expect would have increased the reliability of our results. Unfortunately ADOS research-certified professionals are scarce, and finding such individuals to physically interact with our robot was a major challenge.

Additionally and from a technological point of view, our robot, being autonomous, showed some variability in some behaviors, which may have injected additional noise in our data. Depending on the intended use, the level of autonomy may need to be adjusted depending on the advantages it provides. The embodiment and perceptual capabilities of the robot also constituted limitations to the richness of the interaction. On the one hand, while the robot acted as a reasonable proxy for the child in these simplistic interactions, it could not capture less salient features of child behaviors such as subtle motions and expressions, due to its embodiment. On the other hand, with better perceptual capabilities, the robot could allow for richer interactions, such as picking up on the provider's non-verbal cues. For instance, the robot could recognize gestures such as pointing or changes in gaze directions, which couldn't be reliably implemented with the NAO's moving and low-resolution camera. An external perception solution could mitigate this problem in future iterations.

Finally, as more features and more behaviors are added to our robot in the future, we could incorporate the full pipeline of the ADOS-Sim described in Chapter 4. Such an integration could allow, first, to achieve realistic profiles that reflect real ADOS data (DB-MMS algorithm), and second, to avoid the co-occurrence of incompatible behaviors on the robot (GBS). The behavioral channels may be modified from the previous chapter to fit the NAO robot's modalities specifically (for example, the robot cannot display facial expressions). On the other hand, if robots other than the NAO are used, the behavioral channels as well as the compatibility specifications will have to also be adapted. For instance, a Pepper robot (right side of Figure 5.7), being a mobile robot, allows for smooth location and orientation changes that could be used for behaviors that involve moving around the space. On the other hand, a NAO torso (left side of Figure 5.7) has less mobility than a full body NAO, and cannot use its whole body expressively when gesturing. According to the robot used, the design of individual behaviors will also have to be adapted to accommodate the specific embodiment of the robotic platform.

5.4 Potential impact

The primary application of our customizable robot is to complement therapist training by exposing them to a wide variety of ASD profiles. Beyond complementing therapist training, we also foresee two other potential applications, as reflected by our questionnaire, namely the development of novel therapy tasks, and public education and sensitization. In this section, we discuss the potential impact in each of these applications.

Figure 5.7 Robotic platforms with increasing expressive modalities from left to right.

5.4.1 Therapist training

As discussed in the preamble of this chapter, our contributed methods for enabling interactive robots with customizable 'autism-like' behaviors create opportunities for novel ways of training therapists. First, these robots have the potential to expose therapists to a wide variety of ASD cases, not limited by the availability of real cases. Second, they allow to break down the interaction into parts and focus on them separately, due to the repeatable nature of the interactions. This aspect can allow for safe exploration and refinement of the actions associated with standardized tool administration, potentially under the supervision of a human expert guiding the process. We envision these robotic tools as an interaction-focused complement to existing training methods (video-based coding training, observation of real sessions, etc.).

The contributions presented in this chapter are only the first steps towards a fully functional solution that could be shown to outperform alternative training methods in some circumstances. Towards the development of such deployable solutions, there are a few aspects that we would like to bring up based on our experimentation:

- *Procedural versus coding skills* — An important distinction to make when it comes to desired therapist skills is that of procedural skills (how well the therapist can follow a standardized procedure) versus behavior coding skills (how accurate the therapist is in assigning the correct feature values to a range of observed behaviors).

While behavior coding skills are best trained in a video-based setting that offers the high-fidelity and all the nuances of behavioral expression, we believe that a robot-based training method could mostly benefit the procedural component, in which the robot is a mere proxy for the receiver rather than a high-fidelity replica. Future iterations of a robot-based solution could include a feedback mechanism by which the robot gives positive or negative feedback after every action and every coded behavior, as well as potentially more complex instructions such as explaining the cause of error or prompting the therapist to backtrack or repeat previous actions.

- *Task type* — The ADOS contains a heterogeneous set of tasks with different purposes and amount of structure. In this thesis, we have mostly focused on the most algorithmic tasks, those possessing a clearly defined action scale and a procedure to be followed precisely. Such task types are most suited for a robot-based solution because they are more easily translatable to a programmable scenario, while less structured tasks may be best trained by other methods, for instance through session observation or using actors. More research is needed to better understand the strengths and weaknesses of each training method for different task types.
- *Robotic platform* — In this chapter, we have used a relatively small humanoid robot with plastic casing and minimally anthropomorphic features. It is unclear yet what the "ideal" robotic platform for this type of application looks like, but we suspect that better perception capabilities and more expressive and subtle behaviors may introduce much richer sets of responses. Because the robot is merely seen as a proxy for the child, we suspect that the presence of highly anthropomorphic physical characteristics such as skin or hair is not necessary, or may even be detrimental to the interaction due to the 'uncanny valley' phenomenon [108].

5.4.2 Autism therapy

While therapist training was the main focus of the studies presented in this work, we believe that ASD therapy may also benefit from having a robot capable of exhibiting 'autism-like' behaviors. Specifically, we believe these robots may unlock new possibilities in robot-assisted therapy tasks involving imitation and learning-by-teaching, which we discuss in turn.

Because imitation ability is often impaired in children with ASD [48], imitation tasks hold a special place in ASD therapy [79]. As a result, we believe that an autonomous, customizable, and adaptive robot may be suitable for such tasks. For example, the robot could first match its behavior to that of the child according to their profile, then demonstrate the desirable behavior for the child to imitate. In the context of a long-term interaction, the robot could even evolve towards lower and lower feature values along with the child as progress happens.

On the other hand, such a robot may be used in the context of learning-by-teaching scenarios [97], where a child refines their own skills through teaching the robot previously acquired skills. For example, the robot could be programmed to have slightly lower skills than the child (i.e., higher value on a given a feature), in which case the child teaches the robot to incorporate behaviors that the robot does not have. For example, if the child knows how to make good use of pointing, they could teach a robot that uses only eye gaze to also include pointing in its behavior. However, it is to be noted that a learning-by-teaching approach might be challenging with some children with high ASD severities, and would require empirical investigation.

5.4.3 Education and sensitization

Most available educational resources about ASD for the general public are in the form of written material, although some more interactive resources for children, such as videos[5], have been created. We believe that an interactive and embodied resource, such as our robot, could potentially more efficiently expose people to the different forms and behavioral implications of ASD. Such robots could be used in museums, classrooms, or the workplace, so that people get a better sense of how individuals with different forms of ASD may have very different ways of interacting or responding to social situations. As a result, such a solution could foster a better integration of individuals with ASD in society.

The motivating thoughts of this section do not provide empirical evidence for the usefulness of these applications in the associated contexts, but rather are meant to provide a basis for the conception of future robot-assisted scenarios in the autism domain.

[5] An example of educational autism video can be found at https://bit.ly/2xbJ9mW

5.5 Related work

The use of socially interactive robots as a simulation platform is relatively new, and as such related work on the topic is limited. In this section, we briefly discuss general provider training methods, as well as ones that use physically embodied platforms.

5.5.1 Interactive provider training

Many professions, including those in healthcare and education, require providers to go through a rigorous training in which they develop and refine not only their knowledge but also the practice of certain procedures. Training in an actual real-world situation may often have a high cost associated with it (e.g., mistakes while performing a surgery or flying a plane). These aspects have led many industries to develop solutions to simulate real-world situations and provide tools to train providers to perfect their skills before they apply them in their real professional practice. The paradigm of using simulated environments and interactions for expert training [121, 50] has already been applied to a wide range of fields, including aviation [94], medicine and healthcare [137, 138], the military [103], emergency response [122], and education [87, 10], showing improvement in the performance of trainees in most cases. Simulated environments have also been applied to social settings and interactions [99, 6, 141], as well as procedural tasks [125]. The large majority of these solutions rely on computer simulations and virtual/mixed reality, or virtual agents [86]. However, little work has been done on the introduction of physically embodied agents in these simulated environments.

5.5.2 Physically embodied simulation platforms

In relation to physically embodied forms of interaction, some robots and haptic devices have been used to provide physical feedback and assistance for training dexterous procedures such as surgeries and nursing procedures [137, 78]. In addition to physical interaction, some researchers have started looking at social forms of feedback by simulating social aspects of patient behavior on robots. Of particular interest is a line of work looking at simulating naturalistic pain and other affective behaviors on expressive robotic patient simulators [119, 67]. On the other hand, in an educational context, some approaches have looked at reversing the traditional provider-receiver paradigm through learning-by-teaching [25]. Allowing the student to take on the role of a teacher for a robot or a virtual agent has been shown to benefit their own learning [95, 82].

Even though our work is in line with the above mentioned trends, to the best of our knowledge, the use of social robots in the context of professional therapist training has not been investigated before.

5.6 Summary

In this chapter, we demonstrated an approach on enabling a humanoid robot to exhibit model-based 'autism-like' behaviors of varying severities. We designed 16 behaviors for the NAO robot, based on ADOS descriptions of common behaviors observed in children with ASD. Our robot behaviors spanned different levels of impairment along four selected ADOS features. We integrated those behaviors into an autonomous control architecture, enabling flexible and continuous interactions with humans. Finally, we evaluated our designed behaviors by running a video-based and an 'in situ' study with three trained ASD therapists.

Our results generally show satisfactory levels of accuracy and agreement for most behaviors, although some behaviors may have to be redesigned to reduce the level of subjectivity in coding some robot motions and poses. In particular, estimating gaze direction appeared to be a challenging component of the robot's behaviors. Despite the systematic coding structure of the ADOS, we observed considerable levels of subjectivity in coding for some features. This subjectivity is a known problem in behavior-based diagnostic tools in general [57]. Moreover, as compared to the video-based study, both accuracy and agreement dropped in the real interaction, even though the behaviors of the robot were largely identical. This seems to suggest that the cognitive load of embodied interaction affects the performance of the therapists. These observations therefore motivate the potential use of our solution for complementing therapist training, which currently heavily relies on watching videos. Because current robots can only mimic human behavior in a shallow, exaggerated and simplistic way, an interactive robot capable of simulating simplified versions of a real ADOS interaction may specifically focus on procedural training, as opposed to coding training, for which videos are more adequate.

Our questionnaire results suggest that autism experts are willing to use robotic tools in their professional fields, and holds promise for the use of robots to assist them in their training and practice. The applications we foresee and which were looked at in this research were: complementing therapist training, unlocking novel autism tasks involving robots, and providing interactive tools to educate and sensitize the general population about the diversity of the behavioral aspects of ASD.

Chapter 6

Conclusion and Future Work

This chapter summarizes the contributions made in this thesis and presents some promising directions of future research that build on these contributions. The different elements discussed throughout the thesis are summarized in Figures 6.1 and 6.2, adding more details to Figure 1.1 presented at the beginning of this document.

6.1 Summary of contributions

As presented in Chapter 1, the key contributions of this thesis can be grouped into three categories: algorithmic, methodological, and autism-related.

Algorithmic contributions

- **ADOS-Sim, a simulator that outputs behaviors consistent with high-level children profiles.**

  The two main components of the simulator are the algorithms Descriptor-Based Mean Mapping Sampling (DB-MMS) and Graph-based Behavior Selection (GBS). DB-MMS samples feature vectors informed by an ADOS database of 279 individuals, while GBS selects behaviors consistent with feature vectors while ensuring pairwise compatibility of selected behaviors. The end-to-end ADOS-Sim simulator integrates the two contributed algorithms as well as other components needed to reverse the assessment pipeline.

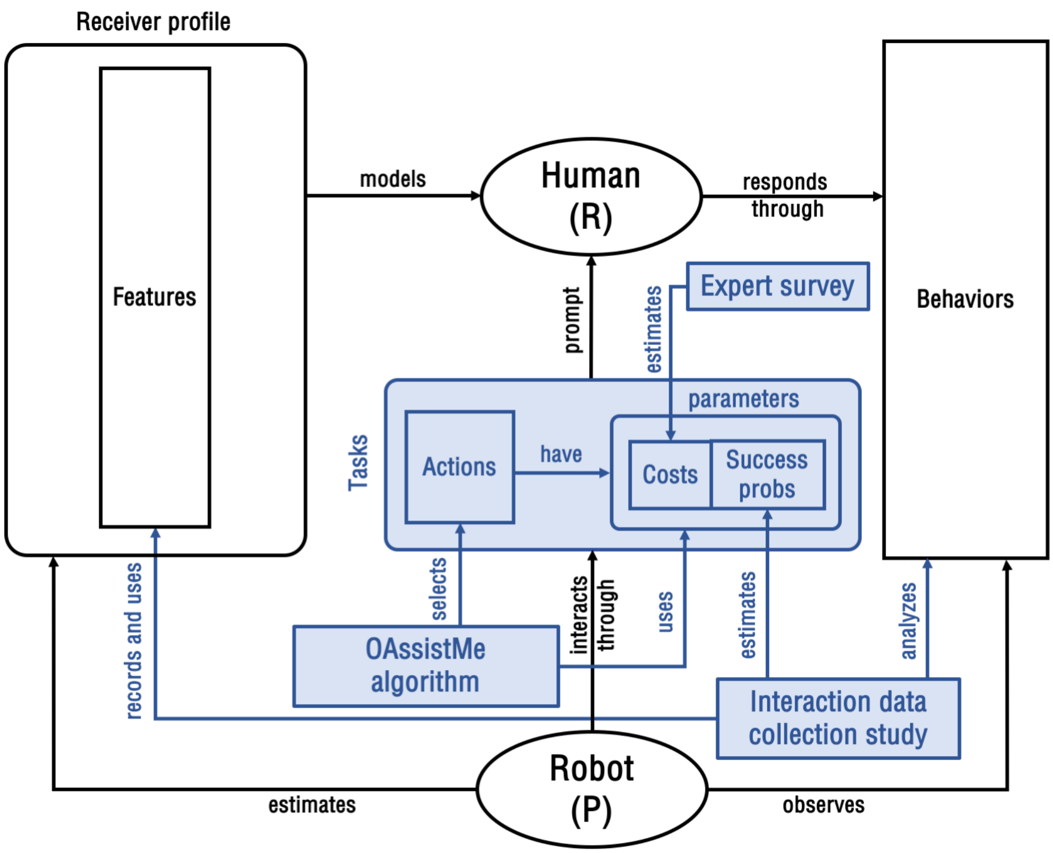

Figure 6.1 Summary of the robot-as-provider approach, explored in this thesis in the context of an interaction between a humanoid robot and a child with ASD (‘P’ refers to provider and ‘R’ to receiver).

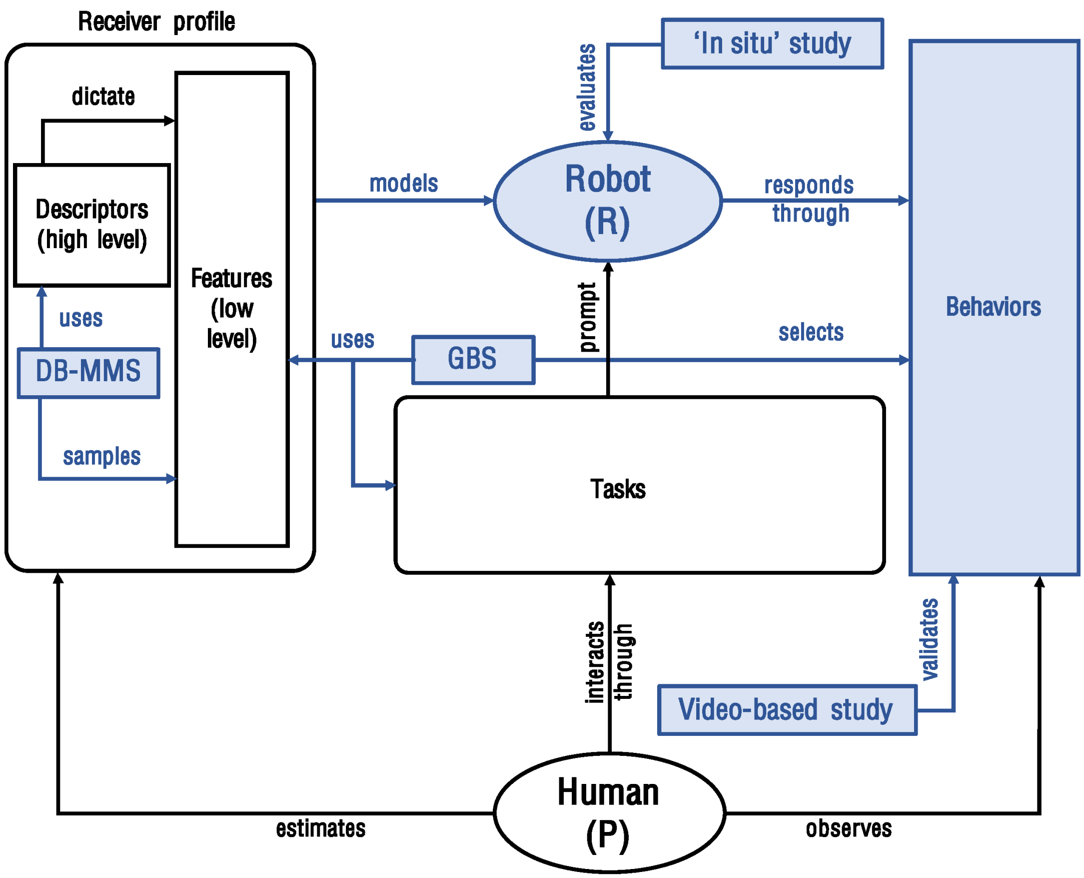

Figure 6.2 Summary of the robot-as-receiver approach, explored in this thesis in the context of an interaction between a humanoid robot and a therapist.

- **OAssistMe, a linear-time algorithm that generates optimal action sequences given action costs and success probabilities.**

  We presented a mathematical formulation of the optimal action sequence generation problem in a general provider-receiver context and in a multi-trial task with: (1) a scale of actions with known and fixed costs and success probabilities, (2) success/failure outcomes at each trial, (3) a horizon corresponding to the maximum allowed number of trials. Based on this formulation, OAssistMe, a linear-time optimal algorithm based on dynamic programming, solves the above problem. We also provided a theoretical analysis of optimal solutions, including proofs of monotonicity and convergence, and constraints on model parameters for suitable algorithm behavior in relation to our application realm.

- **Three extensions of the OAssistMe algorithm that add different assumptions of dependency on the history of actions.**

  The trial-sensitive (TS) extension assumes that the probabilities of success are a function of trial, and are incorporated with a simple modification of the base algorithm. The cost-sensitive (CS) extension assumes that the probabilities are a function of the total cost spent, which makes the revised algorithm slightly more complex. Finally, the repetition-sensitive (RS) assumes that the probabilities are a function of the number of past repetitions of the current action, and can be incorporated using a similar approach to the CS extension. A general version of the algorithm G-OAssistMe was also discussed, as a function of an arbitrary number of features of history.

Methodological contributions

- **A methodology for simulation as inverse-assessment, illustrated using the structure of the ADOS diagnostic tool.**

  The contributed simulation method, at the core of ADOS-Sim, stemmed from the observation that simulation can be seen as an inverse-assessment operation. While assessment maps observations to a single number representing the severity of ASD for a given individual, simulation aims at starting with this severity value and generating consistent behaviors that reflect that value in addition to other descriptors (e.g., age, language ability). We were able to reverse the ADOS assessment pipeline by exploiting its high level of structure and complementing it with data. This simulation method can be applied to other similar structured and standardized assessment tools to help create useful high-level simulations of common receiver responses to standardized tasks.

- **A methodology for the use of robots as receivers to assist the training of providers.**

  We demonstrated an approach on enabling a humanoid robot to exhibit model-based 'autism-like' behaviors of varying severities. We designed 16 behaviors for the NAO robot, based on ADOS descriptions of common behaviors observed in children with ASD. Our behaviors spanned different levels of impairment along four selected ADOS features, namely response to name, response to joint attention, pointing, and language ability. We integrated those behaviors into an autonomous control architecture, enabling flexible and continuous interactions with humans. Finally, we evaluated our designed behaviors by running a video-based and an 'in situ' study with three trained ASD therapists. Our evaluation results generally show satisfactory levels of accuracy and agreement for most behaviors. Our questionnaire results suggest that autism experts are willing to use robotic tools in their professional fields, and hold promise for the use of robots to assist them in their training and practice.

- **A methodology for determining action costs and success probabilities in the context of robot-assisted therapeutic tasks.**

  In the context of a robot-assisted therapeutic interaction involving attention tasks with screens, we presented a methodology for determining action parameters, namely:
  (1) An online survey run with psychologists for determining action costs based on rating the level of explicitness of the robot's actions shown in a video.
  (2) A logistic probabilistic model of children response to robot actions that incorporates action level and child profile. The model was based on data collected during a real interaction between the NAO robot and 10 children with different ASD levels.

Autism-related contributions

- **An analysis of ADOS data from different sources using dimensionality reduction techniques.**

  We obtained heterogeneous data of ADOS feature vectors of 279 individuals, and analyzed them using a combination of dimensionality reduction techniques and clustering. The dimensonality reduction technique allowed us to get an understanding of the distribution of the data points in the ADOS feature space, confirming the presence of a spectrum rather than separate categories of ASD. The clustering algorithm allowed us to generate new data-driven descriptors that could characterize the data in the restricted and full feature space.

- **A scenario based on interactive storytelling, integrating ADOS-inspired tasks addressing specific autism impairments related to attention deficits.**

  We developed a semi-autonomous robotic scenario based on interactive storytelling, which integrates structured tasks inspired by the ADOS tool. The scenario was carefully developed iteratively with the use of pilots and with the feedback of a therapist and a psychologist. The scenario also integrated prompts designed to keep the child engaged, namely prompts to imitate the robot's actions in relation to the story, as well as some questions directed at the child. The scenario was used as a naturalistic interaction context to pursue our research question concerned with the effect of sequencing of the robot's actions on children's response.

- **An exploratory study on the effect of sequencing of a robot's action on the child's response in the context of ADOS-inspired tasks.**

  The study's main goal was to study how different sequencing of a provider robot's actions affects the receiver's response. We studied this problem in the context of two robot-assisted attention-related therapy tasks inspired by the ADOS diagnostic tool. In a first step, we leveraged the structure of the ADOS tasks to build robotic actions on the NAO robot. We then integrated those actions into a control architecture that allows the robot to operate in three modes: Assess, Therapy, and Explore. These modes generate different sequences of the same robot actions, with different properties. Our data collected with 11 children with different ASD severities highlight the advantages and disadvantages of each mode depending on the interaction goals, and was used to build a probabilistic model of child response to the robot's actions.

6.2 Future work

We end this thesis by discussing some promising directions for future work with regard to both the robot-as-provider and robot-as-receiver approaches.

6.2.1 Robot-as-provider

The OAssistMe family of algorithms presented in this thesis consider a fully known and stationary model of the human across task instances. In the real world, the adoption of such robotic solutions will be done in the context of long-term interactions. As a result, the development of more advanced personalized and adaptive algorithms would have to account for and incorporate a dynamic receiver model, potentially not known a priori, or not fully observable. Such an endeavor would have to involve the ability to learn from action outcomes in real time and to reason about perception uncertainty during execution.

6.2.2 Robot-as-receiver

The natural extension of our embodied simulation approach would be to expand it to incorporate additional tasks and features. To do so, we may need to collect additional structured and unstructured behavioral and expert data to inform these developments. Following this development stage, there would be a need for further empirical evidence for the potential benefits of such a robot to complement therapist training, and to determine under which conditions it provides advantages over traditional training methods.

Furthermore, the same methodology could be applied to other healthcare and education domains where standardized interactions are used, such as speech therapy, medical procedures, or educational assessments. Having robots used as an embodied simulated platform opens the path towards having robotic simulators that do not only consider physical aspects of the training, but also social aspects, including communicative and affective behaviors.

6.3 Broader impact and lessons learned

On the one hand, this thesis fell within an established line of work on using robots for therapeutic purposes through interactions with children with ASD focusing on different behavioral, social, and emotional aspects. Specifically, the contributions made in the thesis focused on enabling novel algorithmic tools for allowing increased autonomy and adaptability of a robot's social behavior in assistive settings. The challenges associated with the complexity of a social interaction with a child with ASD make such technological contributions to be key aspects to unlock richer, and potentially more successful, therapeutic scenarios involving socially intelligent robots that can optimize their behavior to help children achieve progress over time.

On the other hand, this thesis made the case for an alternative use of robots in the autism space: that of helping in the training of therapists expected to follow increasingly standardized procedures to assess patients and perform interventions. We believe that social robots have a great potential in penetrating the healthcare sector as embodied simulation platforms that encompass both the physical and the social aspects of an interaction with patients. In the autism space, this embodied aspect of the social interaction is particularly crucial and we hope we are able to encourage both the robotics and clinical psychology communities to tap into this promising research potential in the future.

Some broad but concrete lessons that were learned through our four years of multidisciplinary and collaborative research are listed below, from most specific to broadest.

- **Robots are suitable tools for most but not all children** — There is evidence in the literature that robots are generally attractive to children with ASD, and we have qualitatively observed great levels of excitement towards NAO in our own study. However, we have also observed in our pilots and study that some children are not particularly drawn to the robot, do not sustain their engagement towards it, or show signs of avoidance, especially in the beginning of an interaction. While robots are promising tools for expanding the scope of ASD therapy, one should bear in mind that robots may not be suited for all types of children. For this reason, we see robots as a tool among other tools that therapists can use in the context of therapy.

- **Novel interfaces create novel opportunities but may require additional training** — Therapists expressed positive thoughts, including excitement and surprise, when they interacted with our embodied simulator. However, it was clear from our studies that their performance in coding the robot's behavior dropped from a video setting to a real setting. This observation suggests that, as new technologies are introduced in these professions, proper usability training needs to also be introduced. Any robotic platform will have its own set of limitations (related to perception, cognition, or expressivity). Therefore, having the users be aware of these limitations will help them better adapt to the interface and more fully focus on the purpose of the interaction.

- **We cannot assume that people will always respond to robots the same way they respond to humans** — Our study with children shows that diagnostic information cannot be directly used to predict how children will respond to a robot's actions. For this reason, robot-specific assessment is needed before informing the robot's personalization and adaptation decisions. Furthermore, this discrepancy between response to humans and robots suggests that robots may not be an ideal tool for standardizing ASD diagnosis, as the desired reference measure should be an interaction with a human. In this thesis, we have maintained the human at the center of the diagnosis process while having the robot assist them in potentially improving the accuracy of their diagnostic delivery.

- **Some knowledge may already exist in a different field but it may need 'translation'** — The central methodology adopted in our approach was to leverage existing domain-specific knowledge from a non-computing related field and apply it to a set of computational problems. On the one hand, this approach highlights that there is more useful existing knowledge in other fields than we may initially think when we approach a problem from our field's perspective. On the other hand, it highlights that these models may need some work to be made useful for our research purposes. In our approach, we complemented the ADOS model with some data collected on children interacting with a human and a robot, as well as expert data to quantify qualitative aspects of the model. Other research problems may call for different 'translation' methods, but the latter are nevertheless crucial for bridging the gaps in knowledge across fields.

- **Multidisciplinary research needs multidisciplinary knowledge** — The novel contributions of this research stemmed from an effort to thoroughly understand the problem at hand without solely relying on our inevitably biased pre-assumptions when it comes to the application of interest. This effort involved studying and doing extensive readings about ASD as a developmental disorder, delving into the details and nuances of the ADOS manual, and meeting regularly with autism professionals to validate the usefulness of the ideas put forth in the research. We believe that beyond cross-disciplinary collaboration, multidisciplinary research requires an effort to expand one's knowledge to have a real impact.

6.4 Concluding thoughts

Extrapolating from the scope of this thesis where an assistive robot assumed both the provider and receiver role, we envision a future where robots and humans mutually assist each other and benefit from interacting with one another. These co-assistive human-robot systems would be able to co-learn, co-adapt, and collaborate in a symbiotic way for the benefit of society. Robots are already being deployed in a variety of assistive contexts, both physical and social, often embedded in human-populated environments. As they become more integrated in our society, developing algorithms to support seamless, beneficial, and ethically-aware interactions should be a priority.

Bibliography

[1] Admoni, H., Bank, C., Tan, J., Toneva, M., and Scassellati, B. (2011). Robot gaze does not reflexively cue human attention. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 33.

[2] Ahmad, M., Mubin, O., and Orlando, J. (2017). A systematic review of adaptivity in human-robot interaction. *Multimodal Technologies and Interaction*, 1(3):14.

[3] Alagoz, O., Hsu, H., Schaefer, A. J., and Roberts, M. S. (2010). Markov decision processes: a tool for sequential decision making under uncertainty. *Medical Decision Making*, 30(4):474–483.

[4] Anzalone, S. M., Tilmont, E., Boucenna, S., Xavier, J., Jouen, A.-L., Bodeau, N., Maharatna, K., Chetouani, M., and Cohen, D. (2014). How children with autism spectrum disorder behave and explore the 4-dimensional (spatial 3d+ time) environment during a joint attention induction task with a robot. *Research in Autism Spectrum Disorders*, 8(7):814–826.

[5] Association, A. P. and others" (2013). *Diagnostic and statistical manual of mental disorders (DSM-5®)*. American Psychiatric Pub.

[6] Babu, S., Suma, E., Barnes, T., and Hodges, L. F. (2007). Can immersive virtual humans teach social conversational protocols? In *2007 IEEE Virtual Reality Conference*, pages 215–218. IEEE.

[7] Bagozzi, R. P. (2000). On the concept of intentional social action in consumer behavior. *Journal of Consumer Research*, 27(3):388–396.

[8] Bahadori, S., Cesta, A., Grisetti, G., Iocchi, L., Leone, R. G., Nardi, D., Oddi, A., Pecora, F., and Rasconi, R. (2004). Robocare: pervasive intelligence for the domestic care of the elderly. *Intelligenza Artificiale*, 1(1):16–21.

[9] Baio, J., Wiggins, L., Christensen, D. L., Maenner, M. J., Daniels, J., Warren, Z., Kurzius-Spencer, M., Zahorodny, W., Rosenberg, C. R., White, T., et al. (2018). Prevalence of autism spectrum disorder among children aged 8 years—autism and developmental disabilities monitoring network, 11 sites, united states, 2014. *MMWR Surveillance Summaries*, 67(6):1.

[10] Baptista, M., Martinho, C. R., Lima, F., Santos, P. A., and Prendinger, H. (2014). Improving learning in business simulations with an agent-based approach. *Journal of Artificial Societies and Social Simulation*, 17(3):7.

[11] Baraka, K., Couto, M., Melo, F. S., and Veloso, M. (2019a). An optimization approach for structured agent-based provider/receiver tasks. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, pages 95–103.

[12] Baraka, K., Melo, F. S., Couto, M., and Veloso, M. (2020). Optimal action sequence generation for assistive agents in fixed horizon tasks. *Autonomous Agents and Multi-Agent Systems*, 34(2).

[13] Baraka, K., Melo, F. S., and Veloso, M. (2017a). 'autistic robots' for embodied emulation of behaviors typically seen in children with different autism severities. In *International Conference on Social Robotics*, pages 105–114. Springer.

[14] Baraka, K., Melo, F. S., and Veloso, M. (2017b). Data-driven generation of synthetic behavioral feature vectors modeling children with autism spectrum disorders. In *2017 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 202–208. IEEE.

[15] Baraka, K., Melo, F. S., and Veloso, M. (2017c). Simulating behaviors of children with autism spectrum disorders through reversal of the autism diagnosis process. In *EPIA Conference on Artificial Intelligence*, pages 753–765. Springer.

[16] Baraka, K., Melo, F. S., and Veloso, M. (2018). Towards an embodied simulator of autistic child behaviors: an improved method for selecting simulated behaviors. In *Workshop on Social Robots in Therapy at HRI'18*.

[17] Baraka, K., Melo, F. S., and Veloso, M. (2019b). Interactive robots with model-based 'autism-like' behaviors. *Paladyn, Journal of Behavioral Robotics*, 10(1):103 – 116.

[18] Barakova, E. I. and Chonnaparamutt, W. (2009). Timing sensory integration. *IEEE robotics & automation magazine*, 16(3):51–58.

[19] Barnard, K. E. and Kelly, J. F. (1990). *Assessment of parent-child interaction.* Cambridge University Press.

[20] Barnes, T. and Stamper, J. (2008). Toward automatic hint generation for logic proof tutoring using historical student data. In *International Conference on Intelligent Tutoring Systems*, pages 373–382. Springer.

[21] Bayley, N. (2009). *Bayley-III: Bayley Scales of infant and toddler development.* Giunti OS.

[22] Behrens, T. E., Hunt, L. T., and Rushworth, M. F. (2009). The computation of social behavior. *science*, 324(5931):1160–1164.

[23] Belpaeme, T., Kennedy, J., Ramachandran, A., Scassellati, B., and Tanaka, F. (2018). Social robots for education: A review. *Science robotics*, 3(21):eaat5954.

[24] Bernardo, B., Alves-Oliveira, P., Santos, M. G., Melo, F. S., and Paiva, A. (2016). An interactive tangram game for children with autism. In *International Conference on Intelligent Virtual Agents*, pages 500–504. Springer.

[25] Biswas, G., Leelawong, K., Schwartz, D., Vye, N., and at Vanderbilt, T. T. A. G. (2005). Learning by teaching: A new agent paradigm for educational software. *Applied Artificial Intelligence*, 19(3-4):363–392.

[26] Bomze, I. M., Budinich, M., Pardalos, P. M., and Pelillo, M. (1999). The maximum clique problem. In *Handbook of combinatorial optimization*, pages 1–74. Springer.

[27] Bonani, M., Oliveira, R., Correia, F., Rodrigues, A., Guerreiro, T., and Paiva, A. (2018). What my eyes can't see, a robot can show me: Exploring the collaboration between blind people and robots. In *Proceedings of the 20th International ACM SIGACCESS Conference on Computers and Accessibility*, pages 15–27. ACM.

[28] Brazelton, T. B. and Nugent, J. K. (1995). *Neonatal behavioral assessment scale*. Number 137. Cambridge University Press.

[29] Brunskill, E. and Russell, S. (2011). Partially observable sequential decision making for problem selection in an intelligent tutoring system.

[30] Burgar, C. G., Lum, P. S., Shor, P. C., and Van der Loos, H. M. (2000). Development of robots for rehabilitation therapy: The palo alto va/stanford experience. *Journal of rehabilitation research and development*, 37(6):663–674.

[31] Cabibihan, J.-J., Javed, H., Ang, M., and Aljunied, S. M. (2013). Why robots? a survey on the roles and benefits of social robots in the therapy of children with autism. *International journal of social robotics*, 5(4):593–618.

[32] Cacciabue, P. C., Re, C., and Macchi, L. (2007). *Simple simulation of driver performance for prediction and design analysis*. Springer.

[33] Candes, E. and Recht, B. (2012). Exact matrix completion via convex optimization. *Communications of the ACM*, 55(6):111–119.

[34] Cao, H., Van de Perre, G., Kennedy, J., Senft, E., Esteban, P., De Beir, A., Simut, R., Belpaeme, T., Lefeber, D., and Vanderborght, B. (2018). A personalized and platform-independent behavior control system for social robots in therapy: development and applications. *IEEE Transactions on Cognitive and Developmental Systems*, 8920(c):1–13.

[35] Castro-González, Á., Tan, X. Z., Carter, E., and Steinfeld, A. (2018). Social haptic interaction between robots and children with disabilities. In *Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, pages 81–82.

[36] Chaste, P., Leboyer, M., et al. (2012). Autism risk factors: genes, environment, and gene-environment interactions. *Dialogues Clin Neurosci*, 14(3):281–92.

[37] Chen, G., Warren, J., and Riddle, P. (2010). Semantic space models for classification of consumer webpages on metadata attributes. *Journal of biomedical informatics*, 43(5):725–735.

[38] Chen, J. Y., Lakhmani, S. G., Stowers, K., Selkowitz, A. R., Wright, J. L., and Barnes, M. (2018). Situation awareness-based agent transparency and human-autonomy teaming effectiveness. *Theoretical issues in ergonomics science*, 19(3):259–282.

[39] Clabaugh, C. E., Mahajan, K., Jain, S., Pakkar, R., Becerra, D., Shi, Z., Deng, E., Lee, R., Ragusa, G., and Mataric, M. (2019). Long-term personalization of an in-home socially assistive robot for children with autism spectrum disorders. *Frontiers in Robotics and AI*, 6:110.

[40] Clement, B., Roy, D., Oudeyer, P.-Y., and Lopes, M. (2014). Online optimization of teaching sequences with multi-armed bandits. In *7th International Conference on Educational Data Mining*.

[41] Coeckelbergh, M. (2010). Health care, capabilities, and ai assistive technologies. *Ethical Theory and Moral Practice*, 13(2):181–190.

[42] Conati, C. and Maclaren, H. (2009). Empirically building and evaluating a probabilistic model of user affect. *User Modeling and User-Adapted Interaction*, 19(3):267–303.

[43] Conn, K., Liu, C., Sarkar, N., Stone, W., and Warren, Z. (2008). Affect-sensitive assistive intervention technologies for children with autism: An individual-specific approach. *Proceedings of the 17th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN*, pages 442–447.

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

[45] Corder, G. W. and Foreman, D. I. (2014). *Nonparametric statistics: A step-by-step approach*. John Wiley & Sons.

[46] Dautenhahn, K. and Werry, I. (2004). Towards interactive robots in autism therapy: Background, motivation and challenges. *Pragmatics & Cognition*, 12(1):1–35.

[47] David, D. O., Costescu, C. A., Matu, S., Szentagotai, A., and Dobrean, A. (2018). Developing joint attention for children with autism in robot-enhanced therapy. *International Journal of Social Robotics*, 10(5):595–605.

[48] Dawson, G. and Adams, A. (1984). Imitation and social responsiveness in autistic children. *Journal of abnormal child psychology*, 12(2):209–226.

[49] Defays, D. (1977). An efficient algorithm for a complete link method. *The Computer Journal*, 20(4):364.

[50] Dieckmann, P., Gaba, D., and Rall, M. (2007). Deepening the theoretical foundations of patient simulation as social practice. *Simulation in Healthcare*, 2(3):183–193.

[51] Diehl, J. J., Schmitt, L. M., Villano, M., and Crowell, C. R. (2012). The clinical use of robots for individuals with autism spectrum disorders: A critical review. *Research in autism spectrum disorders*, 6(1):249–262.

[52] Dillon, G. and Underwood, J. (2012). Computer mediated imaginative storytelling in children with autism. *International Journal of Human Computer Studies*, 70(2):169–178.

[53] Edmunds, S. R., Rozga, A., Li, Y., Karp, E. A., Ibanez, L. V., Rehg, J. M., and Stone, W. L. (2017). Brief report: using a point-of-view camera to measure eye gaze in young children with autism spectrum disorder during naturalistic social interactions: a pilot study. *Journal of autism and developmental disorders*, 47(3):898–904.

[54] Epley, N., Waytz, A., and Cacioppo, J. T. (2007). On seeing human: a three-factor theory of anthropomorphism. *Psychological review*, 114(4):864.

[55] Esteban, P. G., Baxter, P., Belpaeme, T., Billing, E., Cai, H., Cao, H.-L., Coeckelbergh, M., Costescu, C., David, D., De Beir, A., Fang, Y., Ju, Z., Kennedy, J., Liu, H., Mazel, A., Pandey, A., Richardson, K., Senft, E., Thill, S., Van de Perre, G., Vanderborght, B., Vernon, D., Yu, H., and Ziemke, T. (2017). How to Build a Supervised Autonomous System for Robot-Enhanced Therapy for Children with Autism Spectrum Disorder. *Paladyn, Journal of Behavioral Robotics*, 8(1):18–38.

[56] Ester, M., Kriegel, H.-P., Sander, J., Xu, X., et al. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, number 34 in 96, pages 226–231.

[57] Falkmer, T., Anderson, K., Falkmer, M., and Horlin, C. (2013). Diagnostic procedures in autism spectrum disorders: a systematic literature review. *European child & adolescent psychiatry*, 22(6):329–340.

[58] Feil-Seifer, D. and Mataric, M. J. (2005). Defining socially assistive robotics. In *Rehabilitation Robotics, 2005. ICORR 2005. 9th International Conference on*, pages 465–468. IEEE.

[59] Folsom-Kovarik, J. T., Sukthankar, G., and Schatz, S. (2013). Tractable pomdp representations for intelligent tutoring systems. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 4(2):1–22.

[60] Forlizzi, J., DiSalvo, C., and Gemperle, F. (2004). Assistive robotics and an ecology of elders living independently in their homes. *Human-Computer Interaction*, 19(1):25–59.

[61] Frankenburg, W., Dodds, J., and Archer, P. (1990). *Denver II.* Denver Developmental Materials, Inc.

[62] Fridin, M. (2014). Storytelling by a kindergarten social assistive robot: A tool for constructive learning in preschool education. *Computers and Education*, 70:53–64.

[63] Friedman, J., Hastie, T., and Tibshirani, R. (2001). *The elements of statistical learning*, volume 1. Springer series in statistics Springer, Berlin.

[64] Gabig, C. S. (2013). Assessment of basic language and learning skills (ablls). *Encyclopedia of Autism Spectrum Disorders*, pages 267–270.

[65] Galitsky, B. (2016). *Computational Models of Autism*. Springer International Publishing, Cham.

[66] Gibbons, P. (2002). *Scaffolding language, scaffolding learning*. Portsmouth, NH: Heinemann.

[67] Gonzales, M., Moosaei, M., and Riek, L. (2013). A novel method for synthesizing naturalistic pain on virtual patients. *Simulation in Healthcare*, 8(6):617.

[68] Gotham, K., Risi, S., Pickles, A., and Lord, C. (2007). The autism diagnostic observation schedule: revised algorithms for improved diagnostic validity. *Journal of autism and developmental disorders*, 37(4):613.

[69] Graham, J. R. (1990). *MMPI-2: Assessing personality and psychopathology*. Oxford University Press.

[70] Greczek, J., Kaszubski, E., Atrash, A., and Matarić, M. (2014). Graded cueing feedback in robot-mediated imitation practice for children with autism spectrum disorders. In *Robot and Human Interactive Communication, 2014 RO-MAN: The 23rd IEEE International Symposium on*, pages 561–566. IEEE.

[71] Grover, S., Chakraborti, T., and Kambhampati, S. (2018). What can automated planning do for intelligent tutoring systems? *ICAPS SPARK*.

[72] Hauskrecht, M. and Fraser, H. (2000). Planning treatment of ischemic heart disease with partially observable markov decision processes. *Artificial Intelligence in Medicine*, 18(3):221–244.

[73] Head, H. (2014). *Aphasia and kindred disorders of speech*, volume 2. Cambridge University Press.

[74] Henkemans, O. A. B., Bierman, B. P., Janssen, J., Neerincx, M. A., Looije, R., van der Bosch, H., and van der Giessen, J. A. (2013). Using a robot to personalise health education for children with diabetes type 1: A pilot study. *Patient education and counseling*, 92(2):174–181.

[75] Hersch, G. I., Lamport, N. K., and Coffey, M. S. (2005). *Activity analysis: Application to occupation*. SLACK Incorporated.

[76] Hoey, J., Boutilier, C., Poupart, P., Olivier, P., Monk, A., and Mihailidis, A. (2013). People, sensors, decisions: Customizable and adaptive technologies for assistance in healthcare. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 2(4):1–36.

[77] Horvitz, E. (1997). Agents with beliefs: Reflections on bayesian methods for user modeling. In *User Modeling*, pages 441–442. Springer.

[78] Huang, Z., Lin, C., Kanai-Pak, M., Maeda, J., Kitajima, Y., Nakamura, M., Kuwahara, N., Ogata, T., and Ota, J. (2017). Impact of using a robot patient for nursing skill training in patient transfer. *IEEE Transactions on Learning Technologies*, 10(3):355–366.

[79] Ingersoll, B. (2008). The social role of imitation in autism: Implications for the treatment of imitation deficits. *Infants & Young Children*, 21(2):107–119.

[80] Ismail, L. I., Hanapiah, F. A., Belpaeme, T., Dambre, J., and Wyffels, F. (2020). Analysis of attention in child–robot interaction among children diagnosed with cognitive impairment. *International Journal of Social Robotics*, pages 1–12.

[81] Itti, L. and Baldi, P. F. (2006). Bayesian surprise attracts human attention. In *Advances in neural information processing systems*, pages 547–554.

[82] Jacq, A., Lemaignan, S., Garcia, F., Dillenbourg, P., and Paiva, A. (2016). Building successful long child-robot interactions in a learning context. In *Human-Robot Interaction (HRI), 2016 11th ACM/IEEE International Conference on*, pages 239–246. IEEE.

[83] Jain, A. K. (2010). Data clustering: 50 years beyond k-means. *Pattern recognition letters*, 31(8):651–666.

[84] Kaiser, S., Träger, D., and Leisch, F. (2011). Generating correlated ordinal random values.

[85] Kenny, L., Hattersley, C., Molins, B., Buckley, C., Povey, C., and Pellicano, E. (2016). Which terms should be used to describe autism? perspectives from the uk autism community. *Autism*, 20(4):442–462.

[86] Kenny, P., Hartholt, A., Gratch, J., Swartout, W., Traum, D., Marsella, S., and Piepol, D. (2007). Building interactive virtual humans for training environments. In *Proceedings of i/itsec*, volume 174, pages 911–916.

[87] Kirkley, S. E. and Kirkley, J. R. (2005). Creating next generation blended learning environments using mixed reality, video games and simulations. *TechTrends*, 49(3):42–53.

[88] Klein, L., Itti, L., Smith, B. A., and Marcelo Rosales, S. N. (2019). Surprise! predicting infant visual attention in a socially assistive robot contingent learning paradigm. In *2019 IEEE International Symposium on Robot and Human Interactive Communication*.

[89] Kohonen, T. (1998). The self-organizing map. *Neurocomputing*, 21(1):1–6.

[90] Kokot, M., Petric, F., Cepanec, M., Miklić, D., Bejić, I., and Kovačić, Z. (2018). Classification of child vocal behavior for a robot-assisted autism diagnostic protocol. In *2018 26th Mediterranean Conference on Control and Automation (MED)*, pages 1–6. IEEE.

[91] Koppula, H. S., Jain, A., and Saxena, A. (2016). Anticipatory planning for human-robot teams. In *Experimental robotics*, pages 453–470. Springer.

[92] Kory Westlund, J. M., Jeong, S., Park, H. W., Ronfard, S., Adhikari, A., Harris, P. L., DeSteno, D., and Breazeal, C. L. (2017). Flat vs. Expressive Storytelling: Young Children's Learning and Retention of a Social Robot's Narrative. *Frontiers in Human Neuroscience*, 11.

[93] Kunda, M., El Banani, M., and Rehg, J. M. (2016). A computational exploration of problem-solving strategies and gaze behaviors on the block design task. In *CogSci*.

[94] Lee, A. T. (2017). *Flight simulation: virtual environments in aviation*. Routledge.

[95] Leelawong, K. and Biswas, G. (2008). Designing learning by teaching agents: The betty's brain system. *International Journal of Artificial Intelligence in Education*, 18(3):181–208.

[96] Leite, I. (2011). Using adaptive empathic responses to improve long-term interaction with social robots. In *International Conference on User Modeling, Adaptation, and Personalization*, pages 446–449. Springer.

[97] Lemaignan, S., Jacq, A., Hood, D., Garcia, F., Paiva, A., and Dillenbourg, P. (2016). Learning by teaching a robot: The case of handwriting. *IEEE Robotics & Automation Magazine*, 23(2):56–66.

[98] Li, L., Chu, W., Langford, J., and Schapire, R. E. (2010). A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 661–670. ACM.

[99] Lin, H. and Sun, C.-T. (2003). Problems in simulating social reality: Observations on a mud construction. *Simulation & Gaming*, 34(1):69–88.

[100] Linebaugh, C. W. and Lehner, L. H. (1977). Cueing hierarchies and word retrieval: A therapy program. In *Clinical Aphasiology: Proceedings of the Conference 1977*, pages 19–31. BRK Publishers.

[101] Lord, C., Rutter, M., Dilavore, P., Risi, S., Gotham, K., and Bishop, S. (2012). *Autism diagnostic observation schedule, second edition*. Western Psychological Services, CA.

[102] Luckin, R., Koedinger, K. R., and Greer, J. (2007). *Artificial intelligence in education: building technology rich learning contexts that work*, volume 158. IOS Press.

[103] Macedonia, M. (2002). Games, simulation, and the military education dilemma. In *Internet and the University: 2001 Forum*, pages 157–167. Educause Louisville, CO.

[104] Madni, A. M. and Madni, C. C. (2018). Architectural framework for exploring adaptive human-machine teaming options in simulated dynamic environments. *Systems*, 6(4):44.

[105] Madsen, L. and Birkes, D. (2013). Simulating dependent discrete data. *Journal of Statistical Computation and Simulation*, 83(4):677–691.

[106] Mardell-Czudnowski, C. and Goldenberg, D. (1998). *DIAL-3: Developmental Indicators for the assessment of Learning*. American Guidance Service Circle Pines, MN.

[107] Melo, F. S., Sardinha, A., Belo, D., Couto, M., Faria, M., Farias, A., Gambôa, H., Jesus, C., Kinarullathil, M., Lima, P., et al. (2018). Project INSIDE: towards autonomous semi-unstructured human–robot social interaction in autism therapy. *Artificial intelligence in medicine*.

[108] Mori, M., MacDorman, K. F., and Kageki, N. (2012). The uncanny valley [from the field]. *IEEE Robotics & Automation Magazine*, 19(2):98–100.

[109] Murray, R. C. and VanLehn, K. (2006). A comparison of decision-theoretic, fixed-policy and random tutorial action selection. In *International Conference on Intelligent Tutoring Systems*, pages 114–123. Springer.

[110] Myers, I. B. (1962). The myers-briggs type indicator: Manual (1962).

[111] Nanavati, A., Tan, X. Z., and Steinfeld, A. (2018). Coupled indoor navigation for people who are blind. In *Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, pages 201–202. ACM.

[112] Nikolaidis, S., Zhu, Y. X., Hsu, D., and Srinivasa, S. (2017). Human-robot mutual adaptation in shared autonomy. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, pages 294–302. ACM.

[113] Palestra, G., Varni, G., Chetouani, M., and Esposito, F. (2016). A multimodal and multilevel system for robotics treatment of autism in children. *Proceedings of the International Workshop on Social Learning and Multimodal Interaction for Designing Artificial Agents - DAA '16*, pages 1–6.

[114] Paradeda, R. B., Ferreira, M. J., Dias, J., and Paiva, A. (2017). How robots persuasion based on personality traits may affect human decisions. In *Proceedings of the Companion of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, pages 251–252. ACM.

[115] Park, H. W., Gelsomini, M., Lee, J. J., and Breazeal, C. (2017). Telling Stories to Robots: The Effect of Backchanneling on a Child's Storytelling. *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, pages 100—-108.

[116] Partington, J. W. (2013). The assessment of functional living skills-the aflstm.

[117] Petric, F., Hrvatinić, K., Babić, A., Malovan, L., Miklić, D., Kovačić, Z., Cepanec, M., Stošić, J., and Šimleša, S. (2014). Four tasks of a robot-assisted autism spectrum disorder diagnostic protocol: First clinical tests. In *IEEE Global Humanitarian Technology Conference (GHTC 2014)*, pages 510–517. IEEE.

[118] Petric, F., Miklic, D., and Kovacic, Z. (2017). Robot-assisted autism spectrum disorder diagnostics using pomdps. In *Proceedings of the Companion of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, pages 369–370. ACM.

[119] Pourebadi, M. and Riek, L. D. (2018). Expressive robotic patient simulators for clinical education. *Proceedings of the R4L Workshop on Robots for Learning Inclusive Learning, at the 13th Annual ACM/IEEE International Conference on Human-Robot Interaction (HRI'18).*

[120] Presecan, M., Petric, F., and Kovacic, Z. (2018). Object classification for child behavior observation in the context of autism diagnostics using a deep learning-based approach. In *2018 26th International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, pages 1–6. IEEE.

[121] Psotka, J. (1995). Immersive training systems: Virtual reality and education and training. *Instructional science*, 23(5-6):405–431.

[122] Querrec, R., Buche, C., Maffre, E., and Chevaillier, P. (2003). Sécurévi: virtual environments for fire-fighting training. In *5th virtual reality international conference (VRIC'03)*, pages 169–175.

[123] Ribeiro, T., Pereira, A., Di Tullio, E., and Paiva, A. (2016). The sera ecosystem: Socially expressive robotics architecture for autonomous human-robot interaction. In *AAAI Spring Symposium Series*.

[124] Ribeiro, T., Vala, M., and Paiva, A. (2012). Thalamus: Closing the mind-body loop in interactive embodied characters. In *International Conference on Intelligent Virtual Agents*, pages 189–195. Springer.

[125] Rickel, J. and Johnson, W. L. (1999). Animated agents for procedural training in virtual reality: Perception, cognition, and motor control. *Applied artificial intelligence*, 13(4-5):343–382.

[126] Rivers, K. and Koedinger, K. R. (2017). Data-driven hint generation in vast solution spaces: a self-improving python programming tutor. *International Journal of Artificial Intelligence in Education*, 27(1):37–64.

[127] Rossi, S., Ferland, F., and Tapus, A. (2017). User profiling and behavioral adaptation for hri: A survey. *Pattern Recognition Letters*, 99:3–12.

[128] Rudovic, O., Lee, J., Dai, M., Schuller, B., and Picard, R. W. (2018). Personalized machine learning for robot perception of affect and engagement in autism therapy. *Science Robotics*, 3(19):eaao6760.

[129] Russell, S. J. and Norvig, P. (2016). *Artificial intelligence: a modern approach*. Malaysia; Pearson Education Limited,.

[130] Scassellati, B. (2003). Investigating models of social development using a humanoid robot. In *Proceedings of the International Joint Conference on Neural Networks, 2003.*, volume 4, pages 2704–2709. IEEE.

[131] Scassellati, B. (2005). Quantitative metrics of social response for autism diagnosis. In *ROMAN 2005. IEEE International Workshop on Robot and Human Interactive Communication, 2005.*, pages 585–590. IEEE.

[132] Scassellati, B., Admoni, H., and Matarić, M. (2012). Robots for use in autism research. *Annual review of biomedical engineering*, 14:275–294.

[133] Scassellati, B., Boccanfuso, L., Huang, C.-M., Mademtzi, M., Qin, M., Salomons, N., Ventola, P., and Shic, F. (2018). Improving social skills in children with asd using a long-term, in-home social robot. *Science Robotics*, 3(21):eaat7544.

[134] Schaaf, R. C. and Roley, S. S. (2006). *Sensory integration: Applying clinical reasoning to practice with diverse populations*. PRO-ED, Incorporated.

[135] Schaefer, A. J., Bailey, M. D., Shechter, S. M., and Roberts, M. S. (2005). Modeling medical treatment using markov decision processes. In *Operations research and health care*, pages 593–612. Springer.

[136] Sevgi, M., Diaconescu, A. O., Tittgemeyer, M., and Schilbach, L. (2016). Social bayes: Using bayesian modeling to study autistic trait–related differences in social cognition. *Biological psychiatry*, 80(2):112–119.

[137] Seymour, N. E. (2008). Vr to or: a review of the evidence that virtual reality simulation improves operating room performance. *World journal of surgery*, 32(2):182–188.

[138] Shapiro, M., Morey, J., Small, S., Langford, V., Kaylor, C., Jagminas, L., Suner, S., Salisbury, M., Simon, R., and Jay, G. (2004). Simulation based teamwork training for emergency department staff: does it improve clinical team performance when added to an existing didactic teamwork curriculum? *BMJ Quality & Safety*, 13(6):417–421.

[139] Shen, Z.-J. M. and Su, X. (2007). Customer behavior modeling in revenue management and auctions: A review and new research opportunities. *Production and operations management*, 16(6):713–728.

[140] Shibata, T. and Wada, K. (2011). Robot therapy: a new approach for mental healthcare of the elderly–a mini-review. *Gerontology*, 57(4):378–386.

[141] Smith, M. J., Ginger, E. J., Wright, K., Wright, M. A., Taylor, J. L., Humm, L. B., Olsen, D. E., Bell, M. D., and Fleming, M. F. (2014). Virtual reality job interview training in adults with autism spectrum disorder. *Journal of Autism and Developmental Disorders*, 44(10):2450–2463.

[142] Steinfeld, A., Jenkins, O. C., and Scassellati, B. (2009). The oz of wizard: simulating the human for interaction research. In *Proceedings of the 4th ACM/IEEE international conference on Human robot interaction*, pages 101–108. ACM.

[143] Sun, M., Leite, I., Lehman, J. F., and Li, B. (2017). Collaborative storytelling between robot and child: A feasibility study. In *Proceedings of the 2017 Conference on Interaction Design and Children*, pages 205–214.

[144] Sundberg, M. L. (2008). *VB-MAPP Verbal Behavior Milestones Assessment and Placement Program: a language and social skills assessment program for children with autism or other developmental disabilities: guide*. Mark Sundberg.

[145] Taheri, A., Alemi, M., Meghdari, A., Pouretemad, H., and Holderread, S. (2015). Clinical application of humanoid robots in playing imitation games for autistic children in iran. *Procedia-Social and Behavioral Sciences*, 176:898–906.

[146] Tapus, A., Peca, A., Aly, A., Pop, C., Jisa, L., Pintea, S., Rusu, A. S., and David, D. O. (2012). Children with autism social engagement in interaction with nao, an imitative robot: A series of single case experiments. *Interaction studies*, 13(3):315–347.

[147] Tartaro, A. (2006). Storytelling with a virtual peer as an intervention for children with autism. *ACM SIGACCESS Accessibility and Computing*, (84):42–44.

[148] Teo, G., Wohleber, R., Lin, J., and Reinerman-Jones, L. (2017). The relevance of theory to human-robot teaming research and development. In *Advances in Human Factors in Robots and Unmanned Systems*, pages 175–185. Springer.

[149] Thill, S., Pop, C. A., Belpaeme, T., Ziemke, T., and Vanderborght, B. (2012). Robot-assisted therapy for autism spectrum disorders with (partially) autonomous control: Challenges and outlook. *Paladyn*, 3(4):209–217.

[150] Trafton, J. G., Schultz, A. C., Perznowski, D., Bugajska, M. D., Adams, W., Cassimatis, N. L., and Brock, D. P. (2006). Children and robots learning to play hide and seek. *Proceeding of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction - HRI '06*, page 242.

[151] Ultsch, A. (1993). *Self-Organizing Neural Networks for Visualisation and Classification*, pages 307–313. Springer Berlin Heidelberg, Berlin, Heidelberg.

[152] Warren, Z. E., Zheng, Z., Swanson, A. R., Bekele, E. T., Zhang, L., Crittendon, J. A., Weitlauf, A. F., and Sarkar, N. (2015). Can Robotic Interaction Improve Joint Attention Skills? *Journal of Autism and Developmental Disorders*, 45(11).

[153] Yussof, H., Salleh, M. H., Miskam, M. A., Shamsuddin, S., and Omar, A. R. (2015). Asknao apps targeting at social skills development for children with autism. In *2015 IEEE international conference on automation science and engineering (CASE)*, pages 973–978. IEEE.

[154] Zander, E., Willfors, C., Berggren, S., Choque-Olsson, N., Coco, C., Elmund, A., Moretti, Å. H., Holm, A., Jifält, I., Kosieradzki, R., et al. (2016). The objectivity of the autism diagnostic observation schedule (ados) in naturalistic clinical settings. *European child & adolescent psychiatry*, 25(7):769–780.

[155] Zhang, Y., Steimle, L., and Denton, B. (2017). Robust markov decision processes for medical treatment decisions. *Optimization online*.

[156] Zheng, Z., Member, S., Young, E. M., Swanson, A. R., Weitlauf, A. S., Zachary, E., Sarkar, N., and Member, S. (2015). Robot - mediated Imitation Skill Training for Children with Autism. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 4320(c):682–691.

Appendix A

Index

This appendix contains a list of acronyms, a guide to the notation used throughout the thesis, and the full list of ADOS features.

Table A.1 List of main acronyms used in this thesis.

Acronym	**Description**
ADOS	Autism Diagnostic Observation Schedule
ASD	Autism Spectrum Disorders
DB-MMS	Descriptor-Based Mean Mapping Sampling
GBS	Graph-based Behavior Selection
HRI	Human-Robot Interaction
ITS	Intelligent Tutoring System
MDP	Markov Decision Process
POMDP	Partially Observable Markov Decision Process
RRB	Restricted and Repetitive Behaviors
SOM	Self-Organizing Map
TD	Typically Developing

Table A.2 Guide to the notation used in this thesis. Bolded variables represent vectors or sets, and starred variables represent optimal quantities.

Notation	Description	Type	Values
a_t	Action taken at trial t. The optimal action for trial t (Chapter 3) is denoted by a_t^*.	Integer	{1,. . . ,N}
β	Weight of logistic function $(1+e^{-\boldsymbol{\beta}\cdot\boldsymbol{\phi}})^{-1}$	Real number	$(-\infty,\infty)$
$b(\hat{f},v)$	Behavior corresponding to feature $\hat{f}$ with a value v	String	Entries of the behavior dataset (English descriptions of behaviors)
$c(a)$	Cost of action a	Real number	$(0,\infty)$
C_t	Cost of action sequence up to t, defined as $\sum_{\tau=1}^{t} c(a_\tau)$	Real number	$(0,\infty)$
D	Descriptor imposing a constraint $\phi(\mathbf{f},D)$ on feature vectors. If more than one descriptor is considered, we denote the set of descriptors by $\mathbf{D}$.	Any	Vary (e.g., single values, class centroids)
$\hat{f}$	Feature name. A full list of features is included in Table A.3.	String	RJA, RNA, LANG, PNT, ...
f	Feature value (higher values mean higher level of impairment). A full list of feature ranges is included in Table A.3.	Positive integer	Vary per feature
$\mathbf{f}$	Feature vector of the form $\langle f_1,\ldots,f_M\rangle$	Vector of integers	Vary according to feature ranges
$\mathbf{h}_t$	History of actions up to but not including t, defined as $\langle a_1,...,a_{t-1}\rangle$	Vector of integers	$\{1,\ldots,N\}^{t-1}$
M	Number of features	Integer	$\{1,\ldots,34\}$
μ	Average (notation used when reporting statistics)	Real number	$(-\infty,\infty)$
n	Number of data points (notation used when reporting statistics)	Integer	$\{1,\ldots,\infty\}$
$n_t(a)$	Number of occurrences of action a in $\mathbf{h}_t$	Integer	$\{1,\ldots,T-1\}$
N	Number of actions	Integer	$\{1,\ldots,\infty\}$
O_T	Expected overall cost for the execution of a given action sequence of length T	Real number	$(-\infty,\infty)$
$p(a)$	Probability of success of action a. The generalized probability function $p(a,\phi_1(\mathbf{h}_t),\phi_2(\mathbf{h}_t),\ldots)$ takes as an input both a and one or more features of the history. Note that the notation p is also used for p-values when reporting statistics.	Real number	$(0,1)$
P_t	Probability that a success first occurs at trial t, defined as $p(a_t)\prod_{\tau=1}^{t-1}(1-p(a_\tau))$	Real number	$(0,1)$
$\mathbf{\Pi}$	Action sequence of the form $\langle a_1,\ldots,a_T\rangle$. An action sequence of length T is denoted by $\mathbf{\Pi}_T$ and an optimal action sequence (Chapter 3) is denoted by $\mathbf{\Pi}_T^*$.	Vector of integers	$\{1,\ldots,N\}^T$
R	Reward associated with the success of an action	Real number	$(0,\infty)$ In applied settings, $R > \min_a c(a)/p(a)$
σ	Standard deviation (notation used when reporting statistics)	Real number	$(0,\infty)$
Σ	ADOS total score $\Sigma=\sum_{i=1}^{M} f_i$	Integer	$\{0,\ldots,28\}$
t	Trial (time step)	Integer	$\{1,\ldots,T\}$
T	Horizon, i.e., length of an action sequence	Integer	$\{1,\ldots,\infty\}$
$\phi(\mathbf{h}_t)$	Feature of the history of actions	Any	Vary (e.g., number of trials t, total cost C_t, number of repetitions $n_t(a)$)
$\psi(\mathbf{f},\mathbf{D})$	Logical constraint on feature vector $\mathbf{f}$, as a function of descriptor values $\mathbf{D}$	Boolean	{True,False}

Table A.3 Full list of features from the ADOS Module 1 [101].

Feature	**Label**	**Value range**
Overall Level of Non-echoed Language	LANG	0–4
Frequency of Vocalization Directed to Others	SVOC	0–3
Intonation of Vocalizations or Verbalizations	IN	0–4
Immediate Echolalia	ECHO	0–3
Stereotyped/Idiosyncratic Use of Words or Phrases	STER	0–3
Use of Other's Body to Communicate	OTHR	0–2
Pointing	PNT	0–3
Gestures	GES	0–2
Unusual Eye Contact	EYE	0–2
Responsive Social Smile	SML	0–3
Facial Expressions Directed to Others	EXPO	0–2
Integration of Gaze and Other Behaviors During Social Overtures	GAZE	0–3
Shared Enjoyment in Interaction	ENJ	0–2
Response to Name	RNA	0–3
Requesting	REQ	0–3
Giving	GIV	0–2
Showing	SHO	0–2
Spontaneous Initiation of Joint Attention	IJA	0–2
Response to Joint Attention	RJA	0–3
Quality of Social Overtures	QSOV	0–3
Amount of Social Overtures/Maintenance of Attention: Examiner	SOVE	0–3
Amount of Social Overtures/Maintenance of Attention: Parent/Caregiver	SOVP	0–3
Quality of Social Response	QSR	0–3
Level of Engagement	ENG	0–3
Overall Quality of Rapport	QRAP	0–3
Functional Play With Objects	PLY	0–3
Imagination/Creativity	IMG	0–3
Unusual Sensory Interest in Play Material/Person	SINT	0–2
Hand and Finger and Other Complex Mannerisms	MAN	0–2
Self-Injurious Behavior	INJ	0–2
Unusually Repetitive Interests or Stereotyped Behaviors	RINT	0–3
Overactivity	OVR	0–2
Tantrums, Aggression, Negative or Disruptive Behaviors	TNT	0–2
Anxiety	ANX	0–2

Appendix B

Proofs and Additional Results

This appendix contains proofs of theoretical results and additional results of the algorithms from Chapter 3.

B.1 Proofs

Our proofs are structured along the following three (mutually exclusive) cases:

(a) $O_1^* > 0$, or equivalently $R < \min_a c(a)/p(a)$.

(b) $O_1^* < 0$, or equivalently $R > \min_a c(a)/p(a)$.

(c) $O_1^* = 0$, or equivalently $R = \min_a c(a)/p(a)$.

Lemma 1: For any T, we have one of:

(a) $0 < O_T^* < \min_a c(a)/p(a) - R$.

(b) $0 > O_T^* > \min_a c(a)/p(a) - R$.

(c) $0 = O_T^* = \min_a c(a)/p(a) - R$.

Proof. We use induction on T. From Equation (3.7):

$$O_T^* = \min_a \{(1 - p(a))O_{T-1}^* + c(a) - p(a)R\}$$

and

$$O_1^* = \min_a \{c(a) - p(a)R\}$$

apply in all cases.

Case (a)

Base case:

$$0 < O_1^* = \min_a c(a) - p(a)R < \min_a c(a)/p(a) - R$$

Induction step:

Assume $0 < O_{T-1}^* < \min_a c(a)/p(a) - R$, then O_T^* is also positive from Equation (3.7) and base case. Also, for all a:

$$\begin{aligned} O_T^* &\le (1 - p(a))O_{T-1}^* + c(a) - p(a)R \\ &< (1 - p(a))(c(a)/p(a) - R) + c(a) - p(a)R = c(a)/p(a) - R \end{aligned}$$

By induction, $0 < O_T^* < \min_a c(a)/p(a) - R$ for all T.

Case (b)

Base case:

$$0 > O_1^* = \min_a c(a) - p(a)R > \min_a c(a)/p(a) - R$$

Induction step:

Assume $0 > O_{T-1}^* > \min_a c(a)/p(a) - R$. Also let $a^\dagger = \arg\min_a c(a)/p(a)$, let a^* be the optimal action selected at stage T, and let $a^{*(1)}$ be the optimal action selected at stage 1.

$$O_T^* \le (1 - p(a^{*(1)}))O_{T-1}^* + c(a^{*(1)}) - p(a^{*(1)})R < 0$$

since $(1 - p(a))O_{T-1}^* < 0$ for any a, and $c(a^{*(1)}) - p(a^{*(1)})R < 0$ (base case).
Also, for all a:

$$\begin{aligned} O_T^* &= (1 - p(a^*))O_{T-1}^* + c(a^*) - p(a^*)R \\ &> (1 - p(a^*))(c(a^\dagger)/p(a^\dagger) - R) + c(a^*) - p(a^*)R \\ &= (1 - p(a^*))c(a^\dagger)/p(a^\dagger) + c(a^*) - R \end{aligned}$$

Using $p(a^*) < p(a^\dagger)c(a^*)/c(a^\dagger)$:

$$\begin{aligned} O_T^* &> \left[1 - p(a^\dagger)c(a^*)/c(a^\dagger)\right] c(a^\dagger)/p(a^\dagger) + c(a^*) - R \\ &= c(a^\dagger)/p(a^\dagger) - R \end{aligned}$$

By induction, $0 > O_T^* > \min_a c(a)/p(a) - R$ for all T.

Case (c)

This case is easily proven by induction on T.

□

Lemma 2: O_T^* is monotonic in T. In particular, it is one of:

(a) strictly increasing, i.e., $O_{T+1}^* > O_T^*$ for all T.

(b) strictly decreasing, i.e., $O_{T+1}^* < O_T^*$ for all T.

(c) constant, i.e., $O_{T+1}^* = O_T^*$ for all T

Proof. Let a^* be the optimal action of stage T.

Case (a)

We have:

$$O_T^*/O_{T-1}^* = 1 - p(a^*) + (c(a^*) - p(a^*)R)/O_{T-1}^*$$

From Lemma 1, for any a:

$$0 < O_{T-1}^* < c(a)/p(a) - R$$

Therefore:

$$(c(a^*) - p(a^*)R)/O_{T-1}^* > p(a^*)$$

We conclude that $O_T^*/O_{T-1}^* > 1$ and $O_T^* > 0$ for all T, which establishes that O_T^* is strictly increasing.

Case (b)

The demonstration that $O_T^*/O_{T-1}^* > 1$ is identical to case (a). Given that $O_T^* < 0$ for all

T, then O_T^* is strictly decreasing.

Case (c)

The result follows from the previous lemma. □

Theorem 3: O_T^* **converges** to $\min_a c(a)/p(a) - R$ as T goes to infinity.

Proof. Lemmas 1 and 2 imply convergence of O_T^* in cases (a) and (b). Furthermore, setting O_{T-1} to O_T in Equation (3.7) results in a single fixed point $\min_a c(a)/p(a) - R$, which establishes the result.

Case (c) is trivial since $\min_a c(a)/p(a) - R = 0$. □

Theorem 4: If $\mathbf{\Pi}^*$ is an optimal sequence, then it is monotonic in t. In particular, $\mathbf{\Pi}^*$ is one of:

(a) **nonincreasing**, i.e., $a_1^* \geq a_2^* \geq ... \geq a_T^*$.

(b) **nondecreasing**, i.e., $a_1^* \leq a_2^* \leq ... \leq a_T^*$.

(c) **constant**, i.e., $a_1^* = a_2^* = ... = a_T^*$.

Proof. In **case (a)**, let a' be an optimal action associated with O_{T-1}^* and a'' an optimal action associated with O_T^*. Then:

$(1 - p(a''))O_T^* + c(a'') - p(a'')R \leq (1 - p(a'))O_T^* + c(a') - p(a')R$

$$(p(a') - p(a''))O_T^* \leq c(a') - c(a'') - R(p(a') - p(a'')) \tag{B.1}$$

and

$(1 - p(a'))O_{T-1}^* + c(a') - p(a')R \leq (1 - p(a''))O_{T-1}^* + c(a'') - p(a'')R$

$$(p(a') - p(a''))O_{T-1}^* \geq c(a') - c(a'') - R(p(a') - p(a'')) \tag{B.2}$$

Combining Equations (B.1) and (B.2), we get:

$$(p(a') - p(a''))O_T^* \leq (p(a') - p(a''))O_{T-1}^*$$

We can conclude that:

If $p(a') > p(a'')$, then $O_T^* \leq O_{T-1}^*$, and if $p(a') < p(a'')$, then $O_T^* \geq O_{T-1}^*$

Assume $a' > a''$. Then $p(a') > p(a'')$, and from the previous result $O^*_T \leq O^*_{T-1}$, which contradicts Lemma 2. Hence, $a' \leq a''$, which establishes that $\mathbf{\Pi}^*$ is nonincreasing.

Similarly, we can show that, in **case (b)**, $\mathbf{\Pi}^*$ is nondecreasing.

In **case (c)**, every step is equivalent to the single trial case, and the same action is selected at every trial, so the resulting sequence is constant. □

B.2 Additional results

Tables B.1, B.2, B.3, and B.4 summarize additional results for the algorithms discussed in Chapter 3 applied to our robot-assisted autism therapy scenario. We include results for the three algorithm extensions (TS, CS, and RS), as well as a version of the algorithm that includes all three features of history (trial, cost, and repetitions), denoted by TCRS. For each algorithm version, we use the logistic regression results accounting for the included features as predictors. For compactness, action sequence $\langle a_1, a_2, \ldots, a_N \rangle$ is reported as $a_1 a_2 \ldots a_N$. As mentioned in Chapter 3, these results are included merely for informative purposes as the algorithm version suitable for our data is OAssistMe, with data from mode Explore only (to avoid large biases in distribution).

Table B.1 Action sequences generated by OAssistMe extensions for $N = 4$ and $T = 6$ (data from **mode Explore**).

Task	**Child profile**	**OAssistMe**	**TS**	**CS**	**RS**	**TCRS**
JATT	1	111124	211114	111111	111111	111111
	2	333334	333334	311111	333333	111111
	3	333344	333334	311111	333333	111111
	4	444444	444444	411111	444444	111111
NAME	1	333334	333334	344111	313244	111111
	2	333344	333444	441111	342344	111111
	3	444444	444444	431111	434231	111111
	4	444444	444444	411111	434231	111111

Table B.2 Action sequences generated by OAssistMe extensions for $N = 4$ and $T = 6$ (data from **modes Explore + Therapy**).

Task	Child profile	OAssistMe	TS	CS	RS	TCRS
JATT	1	111124	112234	344111	123344	311111
	2	333334	333344	444111	324134	311111
	3	333344	444444	444111	342434	111111
	4	444444	444444	441111	434234	111111
NAME	1	333334	444433	411111	312344	111111
	2	333344	444444	411111	342134	111111
	3	444444	344444	411111	342143	311111
	4	444444	333344	411111	432131	211111

Table B.3 Action sequences generated by OAssistMe extensions for $N = 4$ and $T = 6$ (data from **modes Explore + Assess**).

Task	Child profile	OAssistMe	TS	CS	RS	TCRS
JATT	1	111124	233344	441111	333333	111111
	2	333334	333444	441111	333333	111111
	3	333344	344444	441111	444444	111111
	4	444444	444444	411111	444444	111111
NAME	1	333334	311134	111111	313244	111111
	2	333344	333334	311111	342132	311111
	3	444444	444444	331111	342431	111111
	4	444444	444444	333111	432431	311111

Table B.4 Action sequences generated by OAssistMe extensions for $N = 4$ and $T = 6$ (data from **modes Explore + Therapy + Assess**).

Task	Child profile	OAssistMe	TS	CS	RS	TCRS
JATT	1	111124	333334	344411	323344	111111
	2	333334	333444	444111	344344	111111
	3	333344	444444	444111	443434	111111
	4	444444	444444	441111	443434	111111
NAME	1	333334	333344	411111	312344	341111
	2	333344	334444	411111	342134	431111
	3	444444	444444	411111	342143	431111
	4	444444	444444	411111	432131	411111