Learning models of Human-Robot Interaction from small data

Ashkan Zehfroosh,[1] Elena Kokkoni,[1] Herbert G. Tanner,[1] and Jeffrey Heinz[2]

Abstract— **We propose a new approach to learning discrete models for human-robot interaction (HRI) from small data. In this case, HRI is an integral part of a pediatric rehabilitation paradigm that involves a play-based, social environment to improve mobility for young children with such impairments. Applying HRI in this setting is challenging. In order to harness, and potentially even, automate the interaction between the child and the robot, we need some type of an abstract model to capture the causality between the actions of the robot and the reactions of the child. The Markov decision process (MDP) is a model in which the transition probabilities are selected through an empirical approximation procedure called smoothing. Smoothing has been successfully applied in natural language processing (NLP) and identification where, similarly to our paradigm, learning from small data sets is crucial. The goal of this paper is two-fold: 1) to describe our application of HRI, and 2) to provide evidence that supports the application of smoothing for small data sets.**

I. INTRODUCTION

Research on pediatric rehabilitation over the last decade has been focusing on HRI as a way to improve skills in children who faced social and communication challenges [1]–[3]. These studies suggest, for example, that children with autism are able to socially engage in play activities with interactive robots, and even sometimes prefer this type of interaction over that with adults or computer games [4]. Building on this idea, the work of this paper is set in a context where HRI is to be exploited to serve a different population, with relevant objectives. The (early) rehabilitation paradigm here focuses on motor skills. It is novel in using HRI to promote another fundamental skill in early development; that is, the ability to locomote (mobility), for very young children ($<$2 years) with motor impairments.

Early independent locomotion is highly linked to changes in infants' perception, spatial knowledge, social, and language development [5]–[8]. Once infants start moving, they begin to perceive the environment in fundamentally different ways [9]. Populations with delays at their onset of independent locomotion, such as infants diagnosed with Down syndrome [10], [11], have fewer opportunities for self-initiated exploration of their environment and interactions with peers. However, when they are presented with an opportunity for social interaction, they are more responsive compared to children with primarily social impairments, such as children with autism [12]. The hypothesis in this paper is that using HRI with this population may be advantageous in terms of promoting locomotor ability, and potentially self-initiated exploratory and social actions.

To apply HRI in this paradigm, this work involves scenarios that match the infants' abilities and interests based on their age and level of impairment [7], [13], [14]. These scenarios are play activities that may or may not require complex actions from the infants (e.g. climb an inclined platform or a staircase in order to reach and interact with the robot). Since infants are sometimes required to act beyond their level of ability, a body-weight support system was added to potentially assist their movement [13]. One of the primary goals of this early rehabilitation paradigm is to increase the duration and frequency of the infants' locomotor actions by controlled infant-robot interaction. Achieving this interaction in a safe, exciting, and effective manner through automation, brings about an interesting HRI problem.

The HRI problem at hand involves some automated decision-making: what should the robot do to keep the infant moving? In this case, the particular instance of decision-making is arguably more challenging than in other HRI applications primarily for two reasons: (i) the automation system is called to interact with a very young population, and (ii) the automation system is called to effectively work in a much more complex and dynamic environment, compared to that typically used in other HRI pediatric studies [4].

Indeed, infants' behavior and response to automation can vary much more compared to older children and adults. There is significant between- and within-subject variability that may be attributed to the rapid developmental processes during the first two years of human development [15]. However, in this paradigm, the level of variability may be altered by the infants' motor impairments and the complex tasks in the environment they are 'asked' to perform [16]. In this case, exploration of locomotion may produce unpredictable atypical and unique patterns. In addition to the above, available data for such complex interactions are extremely *scarce*. There is a need for developing models that can capture the infants' actions and intentions in such environments, and under these constraints.

For human intent prediction, Markovian models have been successfully applied in other HRI applications. A Markovian model like a Partially Observable Markov Decision Process (POMDP) [17] encodes human intention as hidden states, and treats human actions as observables. A timed POMDP model has been proposed [18] for an automated car to learn socially appropriate behaviors in Pittsburgh-left application scenarios involving human-driven cars. Other related work focuses on updating inaccurate POMDP priors based on observations

[1]Ashkan Zehfroosh, Elena Kokkoni, and Bert Tanner are with the Department of Mechanical Engineering, University of Delaware. {ashkanz, ekokkoni, btanner}@udel.edu

[2]Jeff Heinz is with the Department of Linguistics and Cognitive Science, University of Delaware. heinz@udel.edu

This work was supported by NIH under grant # R01HD87133-01.

using maximum likelihood (ML) [19]. A common thread in these approaches is an underlying assumption on rational human behavior and large amount of observation data. Furthermore, the POMDP model is very demanding in terms of computation (NEXP-complete), and even infinite horizon versions of such problems can be shown to be undecidable under different optimality criteria [20].

A Mixed Observability Markov Decision Process (MOMDP) may present a better option in terms of computational complexity. An MOMDP considers components of states, rather than the whole state, as being unobservable, and has been used for intention-aware robot motion planning [21]. In this application, states of robots and humans are treated as observable, and it is only human intention that is treated as unobservable, assuming that humans follow specific strategies which are known. Optimal policies for robots are then computed by forming a *belief* over human intention. In slightly different instances of applications of MOMDPs in HRI [22], the class of human subjects is considered as the unobservable variable. While it is true that in general MOMDPs drastically decrease the complexity of the problem, they usually require fully known submodels for each one of the unobservable variables. In other words, in an MOMDP formulation if the unobservable variable (human intention, for instance) becomes known, then the whole (human) behavior is determined as well: given the human intention, one can directly predict every human action. Implied in this derivation is that humans are still assumed to reason rationally.

An MDP is a model that may seem simplistic compared to POMDPs and MOMDPs, but it has also been utilized in HRI applications [23], [24]—perhaps not in the degree that the aforementioned models have. Where it lacks in refinement, however, an MDP gains in computational expediency, since it has a smaller number of tunable parameters. This type of model has been used in a multi-user social HRI context [23], and in space applications [24]. In the latter case, the system starts from an inaccurate prior and through observations tries to update the MDP, but because of the size of the model the update process is restricted to select out of a finite set of parameters. When the true values of the model parameters are indeed in that finite set, then the model update will eventually converge to the real model [25].

In the particular application scenario treated in this paper, neither an accurate prior, nor a sufficiently large body of observations can be assumed. For this reason, a process is sought for updating the model in real time, and take into account *every* single observation available at the time where robot action decisions have to be made. With speed and adaptability being the focus, therefore, this paper brings together for the first time in an HRI context an MDP modeling formulation with a computationally efficient machine learning technique called smoothing [26]. Smoothing, a technique traditionally applied in the domain of NLP, is designed to operate and "interpolate" over sparse data sets; it is used here to approximate the unknown parameters of the MDP with an accuracy that a naive ML algorithm is not able to match. The paper demonstrates that it is possible to construct and learn an abstract model of human behavior in the form of an MDP from very small data sets, in a way that shows promise for closing feedback control loops in real-time, computing interaction policies that incorporate knowledge derived from the latest available observations.

II. TECHNICAL APPROACH

This section formalizes an abstract model for HRI in the context of pediatric early mobility rehabilitation, combines this model with a machine learning technique that has been proven successful in the area of NLP from relatively small bodies of text, and finally, it outlines a general approach to behavior planning for the robot.

The model of choice here is an MDP. This is a model of computation that can represent a discrete dynamical system meaningfully that takes a sequence of actions with uncertain outcomes, trying to maximize some notion of utility (its total reward) [21]. This model is deemed appropriate for the following reasons. First, it can abstractly capture important features of HRI in the form of states, actions, and transition between states, in a probabilistic manner that can relate to the uncertainty associated with human behavior. Second, when appropriately designed, this model can be made conveniently simple and abstract, to present the designer with a limited set of parameters that need to be tuned, and in this way facilitate the learning process. The construction of such a model for robot-assisted pediatric rehabilitation is detailed in Section II-A that follows.

In terms of the second component mentioned, the technique of choice is smoothing [26]. Smoothing will be brought to bear to identify the parameters of the MDP, and specifically, its transition probabilities. Those transition probabilities are assumed to capture the infant's action preferences in response to robot actions. The learning algorithm will be updating (on-line) estimates on those transition probabilities, based on observations of robot action and infant action pairs, assuming implicitly some causality between the former and the latter. And although typically one would perform such a parameter update with more conventional methods such as ML and with formal guarantees of convergence [27], the training data size required to obtain convergence is unreasonably big for pediatric rehabilitation applications. In fact, the approximation of the probabilities with maximum likelihood after the (small) amount of observations typically obtained over four to eight clinical sessions are usually very crude and inaccurate. Smoothing, on the other hand, which is a machine learning technique primarily used in NLP [26] to compensate for sparsity in data and give a fair approximation of parameters of the real system, seems to have a better chance of succeeding on small data sets. Details on the adaptation and application of smoothing on MDPs modeling HRI in our pediatric early rehabilitation paradigm are provided in Section II-B. Section II-C is outlining the utilization of the outcome of the learning algorithm for optimally regulating robot behavior aiming at maximizing infant mobility in the form of being robot-triggered or facilitated.

A. Prior model construction

The MDP model is supposed to abstractly encode in its states the possible configurations that infant and robot can find themselves in. These configurations are thought of as the activities that each of the two "players" in this game are engaged to. Each activity, or action, of one player is expected to have a response, or reaction, by the other. For example a robot may move toward the infant, and seeing that the infant may try to get closer, move away, or simply do nothing. Viewing the interaction at this (high) level, and wanting to distinguish between pairs of activities by the corresponding players, indicates that the states in the intended model should encode combinations of activities of infant and robot.

There are therefore two interacting dynamical systems and one is interested in combinations of states of these two systems. This sounds like a parallel composition [28] of two transition systems: one describing evolutions, or sequences of activity transitions on the robot, and another, similar, on the child. Let us therefore define two separate MDPs, one for the robot, M_r and one for the young child M_c, with state and action sets (S_r, A_r) and (S_c, A_c) respectively, and a reward function R from states to reals that encourages behaviors that enhance rehabilitation objectives—for instance, time in motion, or distance traveled—and penalizes inactivity or disengagement. States, therefore, in which the child is moving will naturally produce higher rewards than states where the child stands still. Let the parallel composition of M_r and M_c be denoted M, having state set S and action set A, and reward function be an extension of R onto $S_r \times S_c$.

The underlying transition system of M is what we would call the *arena* (e.g. underlying graph) of a two-player game [29]. It has the structure of a bipartite graph, where players (child and robot) take turns executing transitions. In this system, however, only one type of transitions are controllable: those of the robot. The child's transitions are uncontrollable and in M they are modeled as random ones, with initially unknown transition probabilities. These transition probabilities will be the target of the learning algorithm.

Since there is a fundamental difference between behaviors of different children, there is no basis for assuming that one M model would work for all cases. For each particular subject, therefore, MDP M is initialized with the probabilities associated with the transitions triggered by the child all equal to zero. As observations are made during the course of the rehabilitation sessions, these probabilities will be updated. But initially, the graph of M has several disconnected components, with the states in each (fully) connected component sharing the same child's state (see Fig. 1). There are (controllable) transitions between any two states in a connected component, but no transitions from a state in one component to a state in another.

B. Learning model parameters from observed behavior

Model M is key to effectively regulating the interaction between human and automation during the rehabilitation session. The key idea is to update and refine M on-line based on observations of child's reaction to robot actions. This section describes the approach to updating the parameters of M, i.e., the transition probabilities.

The updating, or learning, process for M adds increasingly more links to the initial group of isolated cliques (see Fig. 1). An intuitive choice of update on the transition probabilities, rooted in ML, is to estimate them as the ratio of times that this particular transition has been observed, over the total number of transitions taken by the child as a response to robot's actions. The problem with such an ML approximation approach to the child's transition probabilities is that it requires a significant amount of data (number of total transitions) in order to achieve adequate degree of convergence. And the amount of data that can be collected from a young child is by default limited; infants develop fast, and after a certain time interval data cease to be representative of the child's particular developmental stage.

Smoothing is designed to compensate for sparsity in a learning data set. Quite common in early applications of NLP, it was shown to interpolate much more effectively compared to other contemporary methods [26]. Motivated by its success in NLP, this paper uses Kneser-Ney smoothing to learn the unknown transition probabilities in MDPs from small sets of observations.

To use Kneser-Ney smoothing, which operates on subsequences of letters, or symbols, the transitions in M will be considered as pairs of states, or bi-gram elements: subsequences of length two. For a transition that takes the system from s_{i-1}, to s_i, for example, the subsequence will be of the form $s_{i-1}s_i$. The learning algorithm thus keeps a record of the frequency of those subsequences that correspond to observed child's transitions. If the frequency of occurrence of a transition from s_i to s_{i-1} upon action a is $c_a(s_{i-1}s_i)$, Kneser-Ney smoothing approximates the probability of actually reaching state s_i upon executing action a as

$$P_{KN}(s_i, a) = \frac{\left|\{s^{'} : 0 < c_a(s^{'}s_i)\}\right|}{\left|\{s's'' : 0 < c_a(s's'')\}\right|} \ , \qquad (1)$$

and assigns probabilities to *all* possible $s_{i-1}s_i$ transitions—i.e., even those that haven't been observed—based on the equation

$$P_a(s_i|s_{i-1}) = \frac{\max\{c_a(s_{i-1}s_i) - \delta, 0\}}{\sum_{s'} c_a(s_{i1}s')} + \lambda_{s_{i-1}}(a) P_{KN}(s_i, a) \qquad (2)$$

in which $\delta \in (0, 1)$ is a constant (discount) parameter, and

$$\lambda_{s_{i-1}}(a) = \frac{\delta}{\sum_{s'} c_a(s_{i-1}s^{'})} \left|\{s' : 0 < c_a(s_{i-1}s')\}\right| \qquad (3)$$

is a normalizing constant that ensures $P_a(s_i|s_{i-1}) \in (0, 1]$.

C. Model-based decision making

The goal of the automated system is to choose a sequence of actions that maximizes the expected total reward $\mathbb{E}\sum_{t=0}^{\infty} \gamma^t R(s_t)$, with γ in the role of a discount factor that reflects the preference of immediate rewards over future ones, and using t to denote the discrete time step, and s_t, a_t the

state and action taken at time t, respectively. A standard method is called Q-learning [30].

Application of Q-learning on MDP M yields an optimal policy $\pi : S \to A$ which maximizes the expected total reward. Given a relatively small MDP representing the child-robot social interaction dynamics, one can update the model and adjust the strategy *on-line*: each time the smoothing algorithm updates the parameters of the MDP, a new optimal strategy can be computed.

An ε-greedy exploration approach balances exploration with exploitation. The robot explores available action options with probability $\varepsilon_t(s)$, and chooses optimal action based on available information with probability $1 - \varepsilon_t(s)$. Then it is ensured [30] that if $\varepsilon_t(s_t) = \frac{c}{N(s_t)}$ with $0 < c < 1$ and $N(s_t)$ denoting the number of times state s_t has been visited, then learning policy will satisfy the "greedy in the limit with infinite exploration" property.

The whole learning and decision-making process described so far is summarized in Algorithm 1.

Input: set of states S, set of actions A, prior transition probabilities $P_a(s_i|s_{i-1})$, reward function $R(s)$, coefficient c.
Set: $N(s) = 0$, $N(s,a) = 0$, $N(s',s,a) = 0$, $\varepsilon_t(s) = 0$, $\forall s \in S$, $\forall a \in A$; current state s_t.
Do

- $N(s_t) \coloneqq N(s_t) + 1$
- $\varepsilon_t(s_t) \coloneqq c/N(s_t)$
- Q-learning (current MDP):
  - with probability $1 - \varepsilon_t(s_t)$: $a_t \coloneqq \arg\max_{a \in A_{s_t}} Q(s_t)$
  - with probability $\varepsilon_t(s_t)$: $a_t \coloneqq \mathsf{Random}(A_{s_t})$
- $N(s_t, a_t) \coloneqq N(s_t, a_t) + 1$
- Observe new state s_n
- $N(s_n, s_t, a_t) \coloneqq N(s_n, s_t, a_t) + 1$
- Update transition probabilities:
  - if maximum likelihood: $P_{a_t}(s_n|s_t) \leftarrow N(s_n, s_t, a_t)/N(s_t, a_t)$
  - if Kneser-Ney smoothing: (2)
- $s_t \leftarrow s_n$

End

Algorithm 1: Learning and decision making loop.

III. Simulation Results

This section presents simulation results of the combined on-line learning and decision-making approach to regulating robot behavior in HRI within a play-based pediatric motor rehabilitation environment. The HRI context is that of a game played between the child and the robot, where each player is trying to "chase" and "catch" the other, and the roles of pursuer and evader switching depending on the distance between child and robot.

A. Early rehabilitation paradigm description

The key insight here is that in order to keep the child engaged and participating, the system has to be responsive, adaptive and excite interest. Thus the robot tries to engage the child in games of chase. One simple game is when the two players start standing on a straight line facing each other. The goal for the robot is to make the child chase it. If the child does not respond the roles reverse: the robot becomes "it" and closes the distance with the child until the child is intrigued to start chasing the robot again. From an algorithmic perspective, the problem is to find out what preferences does the particular child has in this game, and based on these preferences develop game strategies to secure the maximum possible engagement—the latter quantified by periods of time where the human subject were in motion.

B. Model construction and identification

This section illustrates how the prior model of Section II-A for the game of chase can be set up to capture some essential aspects of HRI in the considered application of robot-assisted pediatric rehabilitation, and how this model can be refined based on observations during the course of the particular rehabilitation task: cover a distance of X feet in a straight line, crawling or walking—depending on the developmental stage. The model parameters are identified in two ways, which are then compared to establish the effectiveness and accuracy of each: ML and smoothing.

The prior model of Section II-A consists of two components: the robot's state machine with state set $S_r = \{F, S, B\}$ and action set $A_r = \{f, s, b\}$, and the child's state machine with state set $S_c = \{N, G\}$ and action set $A_c = \{n, g\}$. The semantics of these symbols is as follows: Symbol F stands for the robot moving forward (toward the child), S denotes the state where the robot stands still, and B represents the condition where the robot moves backward (away from the child). Symbols $\{f, s, b\}$ express the robot actions that give rise to transitions to states F, S, and B, respectively. On the human side, symbol G stands for the child making progress toward her goal of X feet of distance, and N for not making progress. Similarly, g and n are thought of as actions taken by the child and resulting in transitions to states G and N, respectively. The parallel composition of these two state machines then gives rise to the transition system of Fig. 1.

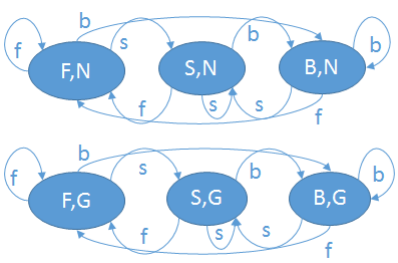

Fig. 1. Prior MDP graph for game of chase between robot and child.

In the system of Fig. 1, the objective is to reach and remain at the states drawn in the lower portion of the figure, sharing

the state component G, which stands for the child making progress toward its rehabilitation task goal. As a result, those states are assigned to higher rewards compared to the ones above sharing the N component. In fact, states (B, G) and (S, G) are relatively more desirable compared to (F, G), because in the latter the robot is "spending its capital," which is its distance to the child, and is thus reducing its future reaction options. The former two states therefore are weighted higher than the latter in the reward function.

The transition probability matrix $P_a(s'|s)$ that represents the chance of jumping from any state s to a state s' upon the robot executing an action $a \in \{f, s, b\}$ is expressed now for a given state, say s_i in the form

$$\begin{bmatrix} P_f(F,N|s_i) & P_s(S,N|s_i) & P_b(B,N|s_i) \\ P_f(F,G|s_i) & P_s(S,G|s_i) & P_b(B,G|s_i) \end{bmatrix} \tag{4}$$

dropping the inner parentheses from states $(\cdot, \cdot)$ to simplify notation. The prior, initial values for the probabilities in (4) are set to

$$\begin{aligned} P(F,N) = P(S,N) = P(B,N) &= \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \\ P(F,G) = P(S,G) = P(B,G) &= \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \end{aligned} \tag{5}$$

with the matrices understood in reference to Fig. 1, having, as elements the probabilities of reaching the particular state from the state in Fig. 1 that is drawn in the same graphical arrangement as that of the element in the matrix. For example, the probability of reaching (F, N) from states (F, N), (S, N) and (B, N) is 1, whereas that of reaching (F, N) from (F, G), (S, G) and (B, G) is zero.

The true, actual transition probabilities are assumed to be

$$\begin{aligned} P(F,N) = P(S,N) = P(B,N) &= \begin{bmatrix} 0.6 & 0.9 & 0.8 \\ 0.4 & 0.1 & 0.2 \end{bmatrix} \\ P(F,G) = P(S,G) = P(B,G) &= \begin{bmatrix} 0.1 & 0.4 & 0.2 \\ 0.9 & 0.6 & 0.8 \end{bmatrix} \end{aligned} \tag{6}$$

but these matrices are unknown to the automation system. The parameters of (6) are used to generate simulated observation data for the learning algorithm, and serve as a standard for comparing the performance of the parameter approximation methods.

C. Model learning

Now Algorithm 1 is employed to refine the prior model of Fig. 1 initialized with (5). Five thousand simulated observations are produced, to form a data sample that is considered here as relatively big. Both ML and smoothing are used to estimate the unknown transition probabilities, and the results are tabulated in Table I.

In general, Table I suggests that both ML and smoothing perform equally well on relatively large data sets. The difference, however, is stark on much smaller data sets. A different test is now conducted on a set of just fifty simulated observations, and the results are listed in Table II.

It is clear from Table II that smoothing outperforms ML on small data sets. With fifty observations the latter has barely changed the inaccurate prior, whereas smoothing has already provided accurate rough approximations of the actual probabilities. As it can be expected, the discrepancy in the respective approximations is reflected in the effectiveness of the decision-making algorithm that reasons based on those estimates. Table III lists the Q-values computed based on the learning outcomes of Table II, and compares them to those that would have been computed if the actual, true model were known a priori. The entries of the Q-matrix express the utility of executing the action a (which indexes the column as $\{f, s, b\}$ respectively) at the state s (which indexes the row as $\{F, N; S, N; B, N; F, G; S, G; B, G\}$ respectively). Despite the apparent roughness of the smoothing approximation in Table II, decision-making based on these learning outcomes leads to optimal action in five out of the six cases. Since in five states, the maximum Q-value occurs at the same action as for the actual model. In comparison, the ML estimates lead to optimal decision only in one out of six cases.

TABLE I

LEARNED MODEL PARAMETERS AFTER 5000 OBSERVATIONS

	Maximum likelihood			Kneser-Ney Smoothing		
$P(F,N)$	0.5698	0.9108	0.8043	0.6263	0.8929	0.8154
	0.4302	0.0892	0.1957	0.3737	0.1071	0.1846
$P(S,N)$	0.5860	0.9105	0.8127	0.6301	0.8984	0.7969
	0.4140	0.0895	0.1873	0.3699	0.1016	0.2031
$P(B,N)$	0.5993	0.9304	0.7719	0.5986	0.8885	0.7778
	0.4007	0.0696	0.2281	0.4014	0.1115	0.2222
$P(F,G)$	0.0946	0.4271	0.2111	0.1141	0.4247	0.2085
	0.9054	0.5729	0.7889	0.8859	0.5723	0.7915
$P(S,G)$	0.1062	0.3970	0.2421	0.1075	0.3834	0.1762
	0.8938	0.6030	0.7579	0.8925	0.6166	0.8238
$P(B,G)$	0.0676	0.4400	0.1835	0.1095	0.4231	0.2066
	0.9324	0.5600	0.8165	0.8905	0.5769	0.7934

TABLE II

LEARNED MODEL PARAMETERS AFTER 50 OBSERVATIONS

	Maximum likelihood			Kneser-Ney Smoothing		
$P(F,N)$	1.0000	1.0000	1.0000	0.8438	0.6667	0.6167
	0.0000	0.0000	0.0000	0.1563	0.3333	0.3833
$P(S,N)$	0.8182	1.0000	0.6667	0.4063	0.8889	0.7125
	0.1818	0.0000	0.3333	0.5938	0.1111	0.2875
$P(B,N)$	1.0000	1.0000	0.5000	0.7031	0.6944	0.7125
	0.0000	0.0000	0.5000	0.2969	0.3056	0.2875
$P(F,G)$	0.0000	0.0000	0.2500	0.1406	0.3611	0.1500
	1.0000	1.0000	0.7500	0.8594	0.6389	0.8500
$P(S,G)$	0.0000	0.0000	0.0000	0.2813	0.1389	0.1000
	1.0000	1.0000	1.0000	0.7188	0.8611	0.9000
$P(B,G)$	0.0000	0.3333	0.5000	0.1406	0.5417	0.1239
	1.0000	0.6667	0.5000	0.8594	0.4583	0.8761

IV. CONCLUSIONS

The results reported provide evidence supporting the development of algorithms with the potential of automating robot-assisted early intervention paradigms that combine mobility and socialization. It is possible to construct an abstract, discrete, model of human behavior incrementally

TABLE III

Q-VALUES AFTER 50 OBSERVATIONS

Models based on	Q-values		
Actual Model	**4.0964**	2.3376	3.8181
	4.1493	2.0402	3.7279
	4.1262	2.2370	3.6254
	5.5992	4.5893	**7.0582**
	5.4711	4.7350	**6.7306**
	5.7833	4.6284	**7.1082**
Kneser-Ney Smoothing	4.0675	4.2799	**5.2671**
	5.4228	2.9256	4.6509
	4.5232	4.0293	4.5206
	6.0414	5.4814	**7.9313**
	5.4722	6.6547	**7.7858**
	6.2522	4.7164	**8.1482**
Maximum likelihood	2.3474	2.6601	**3.9026**
	3.1998	2.4951	**5.2598**
	2.3914	2.5565	**5.7405**
	6.9909	**8.0786**	6.8522
	6.7610	7.8980	**8.0994**
	6.8986	5.9486	5.7857

from small data set, by adapting techniques drawn from natural language processing. Decision-making algorithms based on such methods are capable of outperforming alternative standard techniques for learning the parameters of such discrete models for HRI. The ultimate goal is to incorporate HRI in the field of early mobility rehabilitation and potentially provide high-dosage personalized training in a variety of environments under conditions where human-provided services, although ideal, may not be possible.

REFERENCES

[1] Brian Scassellati, Henny Admoni, and Maja Matarić. Robots for Use in Autism Research. *Annual Review of Biomedical Engineering*, 14:275–294, 2012.

[2] David J Feil-Seifer and Maja J Mataric. Toward Socially Assistive Robotics For Augmenting Interventions For Children With Autism Spectrum Disorders. *Experimental robotics*, 54:201–210, 2009.

[3] Felippe Sartorato, Leon Przybylowski, and Diana K. Sarko. Improving therapeutic outcomes in autism spectrum disorders: Enhancing social communication and sensory processing through the use of interactive robots. *Journal of Psychiatric Research*, 90:1–11, 2017.

[4] Elizabeth S. Kim, Lauren D. Berkovits, Emily P. Bernier, Dan Leyzberg, Frederick Shic, Rhea Paul, and Brian Scassellati. Social robots as embedded reinforcers of social behavior in children with autism. *Journal of Autism and Developmental Disorders*, 43(5):1038–1049, 2013.

[5] J. J. Campos, D. I. Anderson, M. A. Barbu-Roth, E. M. Hubbard, M. J. Hertenstein, and D. Witherington. Travel broadens the mind. *Infancy*, 1(2):149–219, 2000.

[6] M. W. Clearfield. The role of crawling and walking experience in infant spatial memory. *Journal of Experimental Child Psychology*, 89:214–241, 2004.

[7] Karen Adolph. Motor development. *Handbook of child psychology and developmental science*, 2:114–157, 2015.

[8] Eric A Walle and Joseph J Campos. Infant language development is related to the acquisition of walking. *Developmental Psychology*, 50(2):336–348, 2014.

[9] C. Higgins, J. Campos, and R. Kermoian. Effects of self-produced locomotion on infant postural compensation to optic flow. *Developmental Psychology*, 32:836–841, 1996.

[10] Aline Christine Das Neves Cardoso, Ana Carolina de Campos, Mariana Martins Dos Santos, Denise Castilho Cabrera Santos, and Nelci Adriana Cicuto Ferreira Rocha. Motor performance of children with down syndrome and typical development at 2 to 4 and 26 months. *Pediatric physical therapy*, 27:135–41, 2015.

[11] R J Palisano, S D Walter, D J Russell, P L Rosenbaum, M Gémus, B E Galuppi, and L Cunningham. Gross motor function of children with down syndrome: creation of motor growth curves. *Archives of physical medicine and rehabilitation*, 82(4):494–500, 4 2001.

[12] G. Warner, P. Howlin, E. Salomone, J. Moss, and T. Charman. Profiles of children with Down syndrome who meet screening criteria for autism spectrum disorder (ASD): a comparison with children diagnosed with ASD attending specialist schools. *Journal of Intellectual Disability Research*, 61(1):75–82, 2017.

[13] Laura a Prosser, Laurie B Ohlrich, Lindsey a Curatalo, Katharine E Alter, and Diane L Damiano. Feasibility and preliminary effectiveness of a novel mobility training intervention in infants and toddlers with cerebral palsy. *Developmental neurorehabilitation*, 15(4):259–66, 2012.

[14] Karina Pereira, Renata Pedrolongo Basso, Ana Raquel Rodrigues Lindquist, Louise Gracelli Pereira da Silva, and Eloisa Tudella. Infants with Down syndrome: percentage and age for acquisition of gross motor skills. *Research in developmental disabilities*, 34(3):894–901, 3 2013.

[15] Linda Fetters. Perspective on variability in the development of human action. *Physical therapy*, 90(12):1860–7, 12 2010.

[16] Jan P. Piek. The role of variability in early motor development. *Infant Behavior and Development*, 25(4):452–465, 1 2002.

[17] Nikolaos Mavridis. A review of verbal and non-verbal human-robot interactive communication. *Robotics and Autonomous Systems*, 63(P1):22–35, 2015.

[18] Frank Broz, Illah Nourbakhsh, and Reid Simmons. Planning for Human-Robot Interaction in Socially Situated Tasks: The Impact of Representing Time and Intention. *International Journal of Social Robotics*, 5(2):193–214, 2013.

[19] Finale Doshi and Nicholas Roy. Spoken language interaction with model uncertainty: An adaptive human-robot interaction system. *Connection Science*, 20(4):290–318, 2008.

[20] D Bernstein, R Givan, N Immerman, and S Zilberstein. The complexity of decentralized control of Markov decision processes. *Mathematics of operations research*, 27(4):819–840, 2002.

[21] Tirthankar Bandyopadhyay, Kok Sung Won, Emilio Frazzoli, David Hsu, Wee Sun Lee, and Daniela Rus. Intention-Aware Motion Planning. In E. Frazzoli et al., editor, *Algorithmic Foundations of Robotics X*, volume 86, pages 475–491. Springer-Verlag, 2013.

[22] Stefanos Nikolaidis, Keren Gu, Ramya Ramakrishnan, and Julie Shah. Efficient Model Learning for Human-Robot Collaborative Tasks. *arXiv*, pages 1–9, 2014.

[23] Simon Keizer, Mary Ellen Foster, Oliver Lemon, Andre Gaschler, and Manuel Giuliani. Training and evaluation of an MDP model for social multi-user human-robot interaction. *Proceedings of the SIGDIAL 2013 Conference*, pages 223–232, August 2013.

[24] Catharine L R Mcghan, Ali Nasir, Ella M Atkins, Autonomous Aerospace, and Ann Arbor. Human Intent Prediction Using Markov Decision Processes. *AIAA Infotech*, pages 1–16, June 2012.

[25] P. R. Kumar and A. Becker. A New Family of Optimal Adaptive Controllers for Markov Chains. *IEEE Transactions on Automatic Control*, 27(1):137–146, 1982.

[26] Stanley F. Chen and Joshua Goodman. An Empirical Study of Smoothing Techniques for Language Modeling. *Proceedings of the 34th Annual Meeting on Association for Computational Linguistics*, 13:310–318, 1996.

[27] Stuart J. Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. 2010.

[28] Christos G. Cassandras and Stéphane Lafortune. *Introduction to Discrete Event Systems*, volume 2. 2008.

[29] Jane Chandlee, Jie Fu, Kostantinos Karydis, Cesar Koirala, Jeffrey Heinz, and Herbert G. Tanner. Integrating grammatical inference into robotic planning. In *Proceedings of the 11th International Conference on Grammatical Inference*, volume 21, pages 69–83, 2012.

[30] Junling Hu and Michael P Wellman. Nash Q-Learning for General-Sum Stochastic Games. *Journal of Machine Learning Research*, 4(6):1039–1069, 2003.