

Gaining insight into the mechanisms behind a
sense of agency in infancy through babybot
simulation

Author

Dennis Merkus¹
4117638

Supervisors

Lorijn Zaadnoordijk², Maria Otworowska^{1,2},
Johan Kwisthout^{1,2}, Iris van Rooij^{1,2}

¹ Department of Artificial Intelligence, Radboud University

² Donders Institute for Brain Cognition and Behaviour,
Radboud University

August 24, 2015

Bachelor's Thesis in Artificial Intelligence

Abstract

Research on the mechanisms behind a sense of agency is important, as it might give insight into the fundamental structure of intelligent systems. In developmental psychology, infants are hypothesized to develop a sense of agency in the first months of life. By simulating infants using different learning mechanisms, we attempt to gain insight into the conditions under which these mechanisms are distinguishable. This can in turn lead to the specification of which mechanisms are necessary for a sense of agency. The project's goal is to provide a background for research for explaining the experience of a sense of agency. (99 words)

Contents

1	Introduction	1
1.1	Research Questions	4
2	Simulator	5
2.1	Overview	5
2.2	Entity	8
2.3	Mechanism	13
2.4	Formalization of a Simulation	13
3	Mechanisms	15
3.1	Operant Conditioning	15
3.2	Causal Learning	17
4	Results	24
4.1	Normal Condition	24
4.2	Switch Condition	27
5	Discussion	31
5.1	Scientific Challenges	32
6	Conclusion	34

1 Introduction

A sense of agency is something that we as humans take for granted. Whenever one knocks on a door, one feels as if the sound was a result of knocking on the door; a result of one’s action. Such an experience might result from a particular structure or mechanism, as different structures and mechanisms might underly different experiences and even different types of experiences [1, 2]. If we know what kind of mechanism or structure is necessary for an agent to have a sense of agency, then systems can be created with or without such a sense of agency by selecting the appropriate mechanisms and structures in the design of the system [1]. The possibility of creating systems with a sense of agency is of interest to the field of artificial intelligence, as one of the field’s original goals and one of the goals of current artificial general intelligence (AGI) research is to create human-level intelligence [3]. Human-level intelligence does not appear out of thin air. It develops after years of life, going from behavior that is seemingly random, to behavior that is typical of humans such as communication through language and a sense of agency. Then inevitably, during the transition from infancy to adulthood, there are transitions from the absence of such abilities to the presence of those same abilities. Studying these transitions could provide information on these abilities, such as information on the necessary conditions to develop a particular mechanism or information on the characteristics of the mechanisms underlying such abilities. Studying the development of a sense of agency in infancy could be a step in learning about the mechanisms behind a sense of agency and consequently self-awareness.

In the developmental psychology literature, it is claimed that infants develop a sense of agency within the first months of life [4]. An experiment by Rovee-Collier et al. had infants lying in a crib, with a mobile positioned above it. In the experimental group, the right leg was connected to a mobile with a ribbon, whereas in the control group no limbs were connected to the mobile. Connecting the leg to the mobile ensures that the infant can directly control when the mobile moves; whenever the infant kicks, the mobile will move, too. More specifically, the mobile's frequency and intensity of movement is proportionate to the frequency and intensity of the infant's kicks. Whereas infants in the experimental group had control over the mobile's movements, infants in the control group had the mobile's movement controlled by the experimenter who made the mobile move by imitating patterns of kicking produced by infants in the experimental group. Rovee-Collier et al. found that limb activity of subjects in the contingent group changed from activity in all limbs to activity of only the right leg. This was interpreted as infants adopting their limb movements to move in particular the limb that is connected to the mobile, without external supervision.

In later research by Watanabe and Taga involving three-month-old infants, a sense of agency was attributed to this kind of behavior. Here, arm- and leg-based learning were compared. A different pattern of limb movement was observed in each conditions: in the arm-based condition the arms' movement specifically increases, whereas in the leg-based condition all limbs' movement increases. Possible explanations included the fact that the arms' movements are visible to the infant, whereas the legs' movements are not. This was taken to suggest that a sense of agency influences limb movements in the arm-based learning condition [6].

There are multiple possible explanations for the infants' performance, relating differently to the presence or absence of a sense of agency. Two of these mechanisms were chosen for comparison based on what these mechanisms learn. One of these mechanisms is an operant conditioning mechanism that selects actions that are contingent with a rewarding state. If the patterns can be explained by such a mechanism, then this implies that the infants do not necessarily have a sense of agency in the sense described before.

Returning to the definition, a sense of agency is the sense that one's own action causes a change in the world. This requires the ability to recognize changes in the environment, to recognize one's actions, and to perceive relations between changes and these actions. Because the operant conditioning mechanism does not have the ability to recognize changes in the world other than a particular reward and does not have any causal representation, the operant conditioning mechanism does not fulfil the conditions for a sense of agency. Thus, if the previously mentioned experiment's data can be replicated by such an operant conditioning mechanism, then the presence of a sense of agency cannot be inferred from the data alone.

The other mechanism proposed as an explanation for the infants' movement patterns is a causal learning mechanism. This mechanism models the environment and how it changes depending on the actions in a probabilistic model of the world. Because the probabilistic model is a causal model, the condition of causality for the sense of agency is fulfilled. In other words the causal learning mechanism fulfils at least one necessary condition for the sense of agency, whereas the operant conditioning mechanism does not. Then, if the experi-

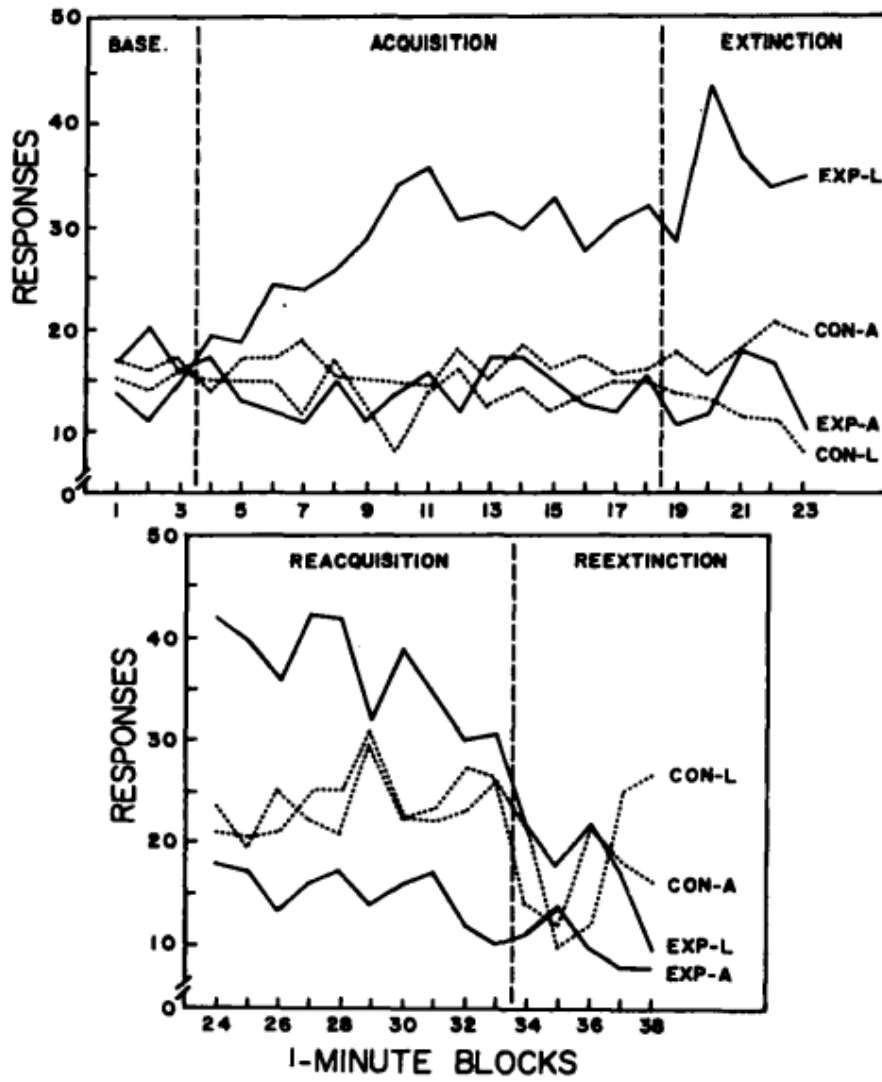


Figure 1: Mean limb movements of right and left arms of 10 infants for the original experiment. EXP-L indicates the connected right foot. (Reproduced with permission from Rovee-Collier et al.)

ment's data can be replicated by the causal learning mechanism but not by the operant conditioning mechanism, then this is an indication that a sense of agency is involved.

1.1 Research Questions

We simulate the infants in the experiment by Rovee-Collier et al. with the previously introduced operant conditioning and causal learning mechanisms, described in more detail later. For convenience, the simulated infants will be called 'babybots' in the rest of this thesis. The babybots will be compared behaviorally with the infants in the original experiment to determine how much the data can be replicated by these mechanisms. A pattern of behavior, or pattern of limb movement, is the frequency of the limbs' movement and the change in these frequencies over time.

Two main questions arise in the comparison of the two mechanisms and the infants. First, can the infants' movement pattern be replicated by the operant conditioning mechanism? Second, can the infants' movement pattern be replicated by the causal learning mechanism?

These two questions might provide evidence to the question of whether a sense of agency could be inferred from the experimental results. If the answer to the first question is yes, that is, the data can be replicated by the operant conditioning mechanism, then no sense of agency can be concluded from the data, because, as mentioned before, the operant conditioning mechanism lacks the capacity to produce representations which are necessary for a sense of agency. If the answer to the first question is no and the answer to the second question is yes, that is, the data can be replicated by the causal learning mechanism, then this is an indication that a sense of agency is involved in producing the infants' behavior. If the answer to both questions is no, then this could mean that the mechanisms should be altered or that a different mechanism is required to produce similar data.

To answer these questions, a simulation environment with a set-up similar to the experiment by Rovee-Collier et al. is designed. An operant conditioning mechanism and causal learning mechanism with the previously mentioned characteristics are designed that are simulated in the simulation environment similar to the experiment. The data from these simulations is then compared to the empirical data.

2 Simulator

To be able to perform simulations within the context of the experiment, a simulator was designed and implemented. Here, an overview of the simulator's components and the motivation behind them is given.

The simulator is created with three points in mind: Firstly, we want to abstract the experimental environment. Abstracting away details that are not directly relevant to the experiment, such as how different components interact, has both advantages and disadvantages. An advantage is that abstracting details can lead to an intuitive description that is easily altered. A major disadvantage is that it is not always obvious to decide which aspects are irrelevant to the experiment and which are not. By abstracting, some complexity of the real situation is lost. This will be discussed while describing the environment for the mobile experiment simulation.

Secondly, the simulator has to be general enough to allow for the use of different kinds of learning algorithms. Because the different algorithms use different representations, in this case based on probability theory, the simulator has to be suited to account for this by having a general representation of the input and output of the algorithms.

Thirdly, the simulator should support data collection and analysis. Specifically in relation to the simulation of the mobile experiment, kicking patterns should be measurable.

To allow for such abstraction, a general, formal description of the simulator is provided. Because the experiment with the babybot and the mobile is of particular interest now, the description of the formalized babybot simulation will be described as an example.

2.1 Overview

The simulator's general structure is illustrated in Figure 2. For a more concrete example describing the environment for the babybot, see Figure 3.

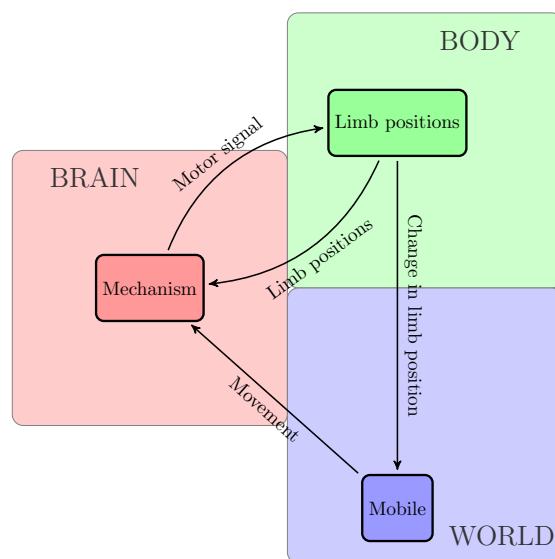


Figure 2: Overview of the simulator's structure. The limb positions change over time. The limb positions are used by the mechanism (i.e. operant conditioning or causal learning) to select new motor signals. When the babybot's limb positions change, this can cause the mobile to move. The mobile sends signals, which are received by the babybot and are used together with the limb positions to select motor signals.

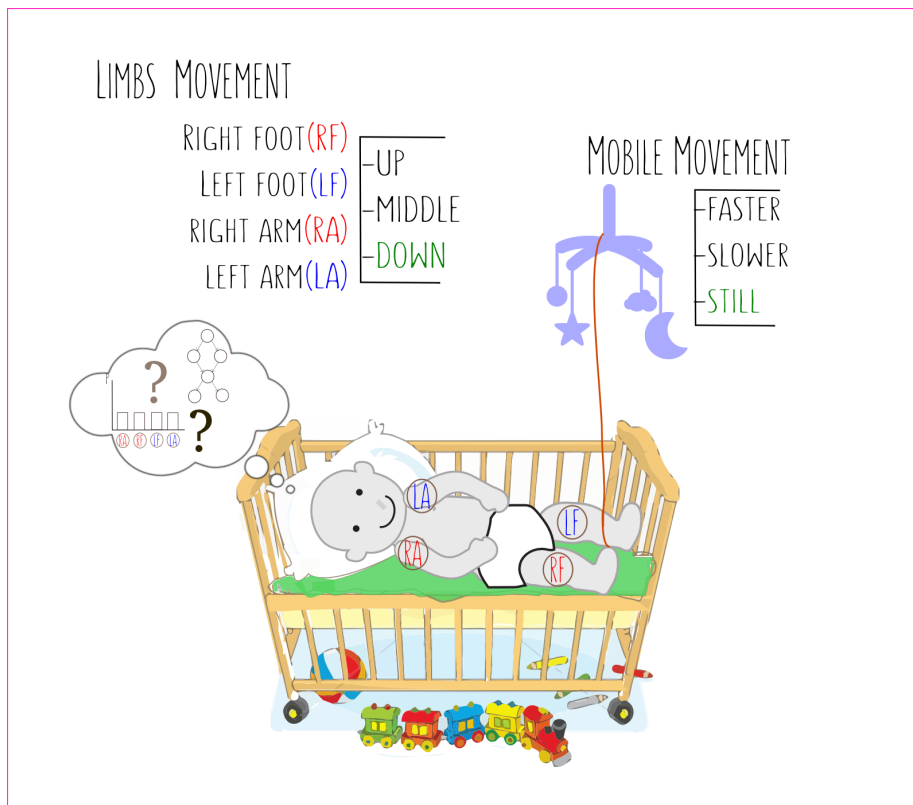


Figure 3: Overview of the simulator environment with a babybot and mobile. The babybot has four limbs, each with three possible positions; it learns by an operant conditioning or causal learning mechanism; and it can observe the mobile's movement, which depends on the mobile's state.

In the experiment, there are two entities: the babybot and the mobile. The babybot has four limbs that each have a position: either ‘up,’ ‘middle,’ or ‘down,’ as depicted by the nodes in Figure 4. A limb’s position can be changed by moving it ‘up’ or ‘down,’ or keeping it ‘still,’ as depicted by the edges in the same figure. These values depicted by the labels are called ‘motor signals.’ Then, if a limb is already in the ‘down’ position, then a motor signal ‘down’ will not change anything and the limb will remain in the same ‘down’ position. Similarly for the ‘up’ position. Given the four limbs and three possible positions for each limb as depicted in Figure 4, the babybot has $3 \cdot 4 = 12$ different possible motor signals.

At any moment, or iteration, in the simulation, all limbs can be moved. For example, all limbs can be moved ‘up’ at one point. It is assumed here that a limb cannot both be moved in more than one direction at the same time, i.e. it is not possible to move ‘up’ and ‘down’ nor ‘still’ and ‘down’ at the same time, etcetera. This results in $3^4 = 81$ combinations of motor signals.

A simulation is described in four parts:

- describing the elements in the world that have properties and that (potentially) interact, referred to as ‘entities;’
- describing the mechanisms that control the entities;
- describing which mechanism selects motor signals for which entity;
- describing how entities affect one another, allowing for interactions.

All parts will be described in more detail in their respective sections.

More formally, a simulation $\mathcal{W} = (E_{\mathcal{W}}, C_{\mathcal{W}}, \gamma_{\mathcal{W}}, L_{\mathcal{W}})$ is an ordered tuple of a set of entity descriptions $E_{\mathcal{W}} = \{\epsilon_1, \dots, \epsilon_{|E_{\mathcal{W}}|}\}$ described in section 2.2 below, a set of mechanism descriptions $C_{\mathcal{W}} = \{c_1, \dots, c_{|C_{\mathcal{W}}|}\}$ described in section 2.3, a relation describing which mechanism selects motor signals for which entity $\gamma_{\mathcal{W}} : E_{\mathcal{W}} \rightarrow C_{\mathcal{W}} \cup \{\emptyset\}$, and a set of links, connecting changes in one entity to changes in another $L_{\mathcal{W}} = \{l_1, \dots, l_{|L_{\mathcal{W}}|}\}$, described later.

In describing which mechanisms select motor signals for which entities, $\gamma_{\mathcal{W}} : E_{\mathcal{W}} \rightarrow C_{\mathcal{W}} \cup \{\emptyset\}$, \emptyset indicates that the entity does not use such a mechanism to select motor signals. For example, in one simulation the babybot’s motor signals are selected by an operant conditioning mechanism, and in another simulation the motor signals are selected by a causal learning mechanism. The mobile does not observe and act, so there is no mechanism selecting motor signals.

2.2 Entity

The first main component in the simulator is the ‘entity.’ A distinction has to be made between the description of an entity, ϵ , and an actual instantiation e : the *description* describes how an *instantiation*’s state *changes* over time and under which conditions. In other words, the description does not have a state and does not change, whereas the instantiation does.

An entity description is of the form $\epsilon = (\mathcal{F}, D_{\mathcal{F}}, \mathcal{A}, D_{\mathcal{A}}, \mathcal{M}, r, EC, T, \text{init})$. Such a description contains a set of attribute symbols $\mathcal{F} = \{f_1, \dots, f_{|\mathcal{F}|}\}$, $\mathcal{F} \neq \emptyset$ and a set of domains of values associated with each attribute symbol $D_{\mathcal{F}} = \{D_{f_1}, \dots, D_{f_{|\mathcal{F}|}}\}$ and each attribute f_i has at least one possible value $D_{f_i} \neq \emptyset$.

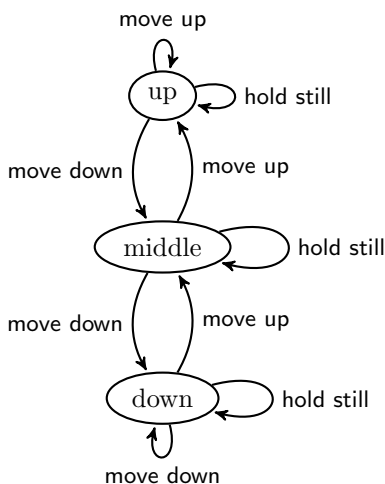


Figure 4: Finite state machine depicting the dynamics for a single limb. Nodes represent positions and edges represent motor signals. An edge with motor signal s from one node a to another node b depicts how the previous position a results in the current position b given motor signal s . Note that an ‘up’ motor signal in the ‘up’ position results in the same ‘up’ position and similarly for the ‘down’ position and the ‘down’ motor signal.

The babybot has four limbs, which are modelled by their positions, $\mathcal{F}_{\text{babybot}} = \{\text{lhs}, \text{rhs}, \text{lhs}, \text{rhs}\}$, i.e. ‘left hand position,’ ‘right hand position,’ ‘left foot position,’ and ‘right foot position’ respectively, abbreviated for convenience. These limbs can all be in one of three positions, $D_{\text{lhs}} = D_{\text{rhs}} = D_{\text{lhs}} = D_{\text{rhs}} = \{\text{down}, \text{middle}, \text{up}\}$ where *down* means the limb is on the surface, *up* means the limb is at maximum range towards the mobile, and *middle* means the limb is somewhere in-between *up* and *down*.

The mobile moves ‘forward’ or ‘backward’ with a certain velocity, and has attributes $\mathcal{F}_{\text{mobile}} = \{\text{vel}, \text{pos}, \text{dir}\}$, where the velocity ranges from 0 to 10 $D_{\text{vel}} = [0, 10]$, the position also ranges from 1 to 10 $D_{\text{pos}} = [1, 10]$ and there is a positive and negative direction $D_{\text{dir}} = \{+, -\}$.

Attributes can combine with a value from the respective domain, resulting in the set of all possible attribute assignments $\{(f, d) \mid f \in \mathcal{F}, d \in D_f\}$. Some example attribute assignments are (lhs, up) and $(\text{vel}, 3)$.

The function $\text{init} : f \rightarrow D_f$ describes the initial value for each attribute.

The infant’s initial condition is one where its limbs are in the middle between the ‘down’ and ‘up’ positions:

$$\begin{aligned}
 \text{init}_{\text{babybot}}(\text{lhs}) &= \text{middle} \\
 \text{init}_{\text{babybot}}(\text{rhs}) &= \text{middle} \\
 \text{init}_{\text{babybot}}(\text{lhs}) &= \text{middle} \\
 \text{init}_{\text{babybot}}(\text{rhs}) &= \text{middle}
 \end{aligned}$$

This position is chosen over the more ‘natural’ position with all limbs down that would be expected from infants, as such a ‘middle’ starting position provides for less skew in the initial positions. A different initial position could be one where all limb positions are randomized, accounting for movement before the

experiment starts. However, because algorithms are being compared, it seems beneficial to control for these different initial positions.

The initial state of the mobile, is one where the mobile does not move:

$$\begin{aligned}\text{init}_{\text{mobile}}(\text{vel}) &= 0 \\ \text{init}_{\text{mobile}}(\text{pos}) &= 5 \\ \text{init}_{\text{mobile}}(\text{dir}) &= +\end{aligned}$$

Similar to the attributes, the set of motor signal symbols $\mathcal{A}_\epsilon = \{\alpha_1, \dots, \alpha_{|\mathcal{A}|}\}$ and respective domains $D_{\mathcal{A}} = \{D_{\alpha_1}, \dots, D_{\alpha_{|\mathcal{A}|}}\}$, $D_{\alpha_i} \neq \emptyset$ can combine to form motor signals $\{(\alpha, d) \mid \alpha \in \mathcal{A}, d \in D_\alpha\}$.

All the babybot's limbs can move, $\mathcal{A}_{\text{babybot}} = \{\text{lh, rh, lf, rf}\}$, i.e. 'left hand,' 'right hand,' 'left foot,' and 'right foot,' in the direction 'up' or 'down,' or held 'still' in the position the limb is already in $D_{\text{lh}} = D_{\text{rh}} = D_{\text{lf}} = D_{\text{rf}} = \{\text{up, down, still}\}$. Note that it is possible to have an entity without motor signals, or in other words, an entity that cannot 'act.' The mobile is one such entity, $\mathcal{A}_{\text{mobile}} = \emptyset$. A combination of motor signals σ , is a set of motor signals with all motor signal symbols appearing exactly once, $\sigma = \{(\alpha_1, d_1), \dots, (\alpha_{|\mathcal{A}|}, d_{|\mathcal{A}|})\}$ with $\alpha_i \in \mathcal{A}$ and $d_i \in D_{\alpha_i}$. The enumeration of all possible combinations form the set Σ .

Now, let \mathcal{E}_ϵ be the set of possible instantiated entities from a description ϵ . An entity instance $e_\epsilon \in \mathcal{E}_\epsilon$, or entity for short, is a set of attribute assignments $e_\epsilon = \{(f_1, d_1), \dots, (f_{|\mathcal{F}|}, d_{|\mathcal{F}|})\}$ for $f_i \in \mathcal{F}_\epsilon$, $d_i \in D_{f_i}$.

2.2.1 Changing an entity's state

The values of an instantiated entity's attribute assignments can change in three ways. First, a rule in the entity description $r : \mathcal{E} \rightarrow \mathcal{E}$ describes how an entity's attributes changes over time, by specifying how the attribute assignments at time t are a function of attribute assignments at time $t - 1$. An example of such a change for velocity and acceleration attributes, denoted by *vel* and *acc* respectively, could be:

$$r([\text{acc} = a, \text{vel} = v]) = [\text{acc} = a, \text{vel} = a + v]$$

The mobile has several such rules. Over time, the velocity decreases until it reaches 0, $r_{\text{vel}}([\text{vel} = v]) = [\text{vel} = v - 1]$. The position changes as a function of the velocity, where the direction is reversed at the extremes as if bouncing between two walls to get movement that resembles a simplified pendulum. Take p to be the mobile's current position, then in the case where $0 < p + v < 10$:

$$r_{\text{pos}}([\text{pos} = p, \text{dir} = d]) = [\text{pos} = p + v, \text{dir} = d]$$

and in the case where $p + v \leq 0$ or $p + v \geq 10$:

$$r_{\text{pos}}([\text{pos} = p, \text{dir} = d]) = [\text{pos} = \text{abs}(p + v), \text{dir} = \text{inv}(d)]$$

where $\text{inv}(-) = +$ and $\text{inv}(+) = -$.

Second, whenever a motor signal is sent, the entity's resulting state is given by the function in the entity description $\alpha : \mathcal{E} \times (\mathcal{A} \times D_{\mathcal{A}}) \rightarrow \mathcal{E}$ that computes

the new attributes' values from an instantiation and an action, i.e. change in state. For example,

$$\alpha([\text{vel} = v, \text{acc} = a], [\text{acc} = x]) = [\text{vel} = v, \text{acc} = a + x]$$

The babybot sends motor signals that in turn change the babybot's limbs' positions given certain constraints, as illustrated in Figure 4. If a limb is either already in the lowest position and moved down, or in the highest position and moved up, then the position for that limb does not change. Yet, in all other cases, the change in position corresponds to the change defined by the motor signal. Formally:

$$\begin{aligned} \alpha_p(a = \text{up}, p = \text{up}) &= \text{up} \\ \alpha_p(a = \text{down}, p = \text{down}) &= \text{down} \\ \alpha_p(a = \text{still}, p = p') &= p' \\ \alpha_p(a = \text{up}, p = \text{down}) &= \text{middle} \\ \alpha_p(a = \text{up}, p = \text{middle}) &= \text{up} \\ \alpha_p(a = \text{down}, p = \text{up}) &= \text{middle} \\ \alpha_p(a = \text{down}, p = \text{middle}) &= \text{down} \end{aligned}$$

where $p \in \mathcal{F}$, $p' \in D_p$ and $a \in \mathcal{A}$ and a and p are related so that they represent the same limb, i.e. if $p = \text{lh}$ then $a = \text{la}$ and so forth. For all $p \in \mathcal{F}$, $a \in \mathcal{A}$ where p and a are for different limbs, the position does not change:

$$\alpha_p(a = a', p = p') = p'$$

for all $a' \in D_a$, $p' \in D_p$.

Third, if a trigger was activated, this might change an entity's attributes in a way similar to a rule, $\tau_\epsilon : \mathcal{E} \times T \times D_T \rightarrow \mathcal{E}$. A trigger $t \in T$, $T = \{t_1, \dots, t_{|T|}\}$ with values $D_T = \{D_{t_1}, \dots, D_{t_{|T|}}\}$, $D_{T_i} \neq \emptyset$ is a symbol that can have a value with it, similar to attributes. An example for activation of a trigger for a ball:

$$\tau([\text{vel} = v], \text{kicked}, \text{forward}) = [\text{vel} = v + 5]$$

Whenever the mobile is moved, in this case by the babybot, the velocity is increased by some fixed amount Δv if the movement is in the same direction, and decreased otherwise. If movement is down and the direction is negative: $t_{\text{vel}}([\text{vel} = v, \text{dir} = -], \text{moved}, \text{down}) = [\text{vel} = \min(10, v + \Delta v), \text{dir} = -]$. If movement is in the opposite direction, i.e. the mobile's movement is positive, then the velocity is set to the negative velocity instead: $t_{\text{vel}}([\text{vel} = v, \text{dir} = +], \text{moved}, \text{down}) = [\text{vel} = \Delta v, \text{dir} = -]$.

2.2.2 Changing another entity's state

The first way in which one entity's attributes can change another entity's attributes, is through sending and receiving signals, such as a beam of light

or a sound. These signals are considered to be in some kind of modality, such as light, sound or smell, and are assumed to move through space at a certain speed. Given the set of possible modalities $\mathcal{M} = \{m_1, \dots, m_{|\mathcal{M}|}\}$, the set of signal types for each modality $S_m = \{S_{m_1}, \dots, S_{m_{|\mathcal{M}|}}\}$, $m_i \in \mathcal{M}$ where $S_{m_i} = \{s_{m_i^1}, \dots, s_{m_i^{|S_{m_i}|}}\}$, and the set of values for every signal $D_{\mathcal{M}} = \{D_{s_{m_1^1}}, \dots, D_{s_{m_{|\mathcal{M}|}^{|S_{m_{|\mathcal{M}|}}|}}}\}$, then a signal (m, s, d) , $m \in \mathcal{M}$, $s \in S_m$, $d \in D_s$ describes the modality, type and value of the signal. When a signal is emitted, its modality is checked against all entities' modalities and added to the observations O of an entity e if the entity can sense the modality, i.e. $m \in \mathcal{M}_{e_e}$.

The infant in the experiment is assumed to be able to see the mobile's movement, so $\mathcal{M}_{\text{babybot}} = \{\text{*sight*}\}$. The mobile does not have eyes, $\mathcal{M}_{\text{mobile}} = \emptyset$.

The sensing of signals could have been modelled as changes in attributes instead, i.e. similar to a rule or action $\mathcal{E}_e \times \Sigma \rightarrow \mathcal{E}_e$. However, the current representation allows a bit more freedom in designing and describing an environment, as designing sensors and signal types is more modular than specifically designing all attributes for a particular entity. Still, it is unclear how important modularity is in the design of this simulator for the mobile experiment in particular, compared to potential problems that this representation brings. For example, one disadvantage of this approach is that there is a conceptual distinction between sensing the other entities, and sensing the entity's own attributes. However, this problem is solved in the interaction between an entity and its action-selecting component, as described below.

Signals are emitted by checking the emission $\eta : e \rightarrow (\mathcal{M} \times S \times D_S)^{\mathbb{N}_0}$ which returns a set of signals, if any, depending on the entity's attribute values. The babybot does not send any such signals, as there are no other entities to receive them. The mobile sends signals indicating changes in its movement. These are observed by the babybot and are what the babybot uses to learn. The mobile can send three possible signals:

- (movement, faster)
- (movement, slower)
- (movement, none)

Which are sent respectively when the mobile's current *vel* is higher than the previous *vel*, lower than or equal to the previous *vel*, and when *vel* is 0.

The second way in which one entity's attributes can change another entity's attribute values, is by activating triggers through events. By having a ('physical') link between two entities, these changes can be triggered. A link $l \in L$ is a tuple $l = (e_c, f, t, e_a)$ describing which entity e_c 's change in attribute f causes the trigger t to trigger in affected entity e_a . The value that is sent together with the trigger is calculated from the change in the causing entity's attribute's value with a function $\Delta_{\mathcal{F}} : \mathcal{F} \times D_f \rightarrow D_f$ where f is the domain of the first parameter.

Although this resembles signals, triggers are mostly physical of nature, such as pushing and pulling of objects, whereas signals were considered to be transmitted through air. Moreover, next to having descriptions of events and triggers in entities, there is a description to connect them in the world description directly.

For example, there might be an infant that can push forward and a ball that rolls when pressure is applied, but if the infant is not in front of the ball then the ball will not roll regardless of the infant pushing.

In the original experiment, the infant’s right leg is connected to the mobile by a ribbon. Up-down movement of the leg results in movement of the mobile through the connection made by the ribbon. Similarly, in the simulator, right leg movements trigger movement of the mobile. This can be accomplished by a link between the babybot’s right foot and the mobile’s move trigger,

$$l_{\text{ribbon}} = (\text{babybot}, \text{rfp}, \text{moved}, \text{mobile})$$

2.3 Mechanism

The second main component is the mechanism, which is a ‘black box,’ receiving observations as input and giving motor signals as output at every time step. A mechanism $c = (M, \text{ACT})$ has some kind of (minimal) memory representation M that is used by the mechanism’s algorithm ACT and can be empty, and a function or algorithm $\text{ACT} : M \times O \times \Sigma \rightarrow \Sigma$ that returns one of the possible motor signal combinations given a set of observations.

Descriptions and pseudo code of the specific algorithms used can be found in Section 3.

2.4 Formalization of a Simulation

To run a simulation, a world description \mathcal{W} , and the entities’ initial states, describing values for all attributes, are required. The algorithm that is used for the simulation is described in Algorithm 1.

The world state W_0 is initialized by instantiating the entities with the initial states, after which the mechanism’s memory is initialized with its associated entity’s state. This results in world state W_0 . A world state at time t is $W_t = (E_t, C_t)$ describes the entities’ and mechanisms’ states at any time point.

From this initial state a simulation proceeds sequentially, updating the next state W_{t+1} from the latest state W_t . Every iteration, the same steps are computed. First, the entities’ rules are processed, which change the entity’s attributes. Second, entities that have triggers are changed by computing their effects. Third, signals that are emitted in the current iteration are checked with the entities that have the appropriate modality. Fourth, motor signals are selected by entities with mechanisms. Then, the results of those selected motor signals are computed.

Algorithm 1 Running the simulation

```
procedure RUNSIMULATION( $\mathcal{W}, E_0, n$ )  
   $\triangleright \mathcal{W}$ , the world description  
   $\triangleright E_0$ , entities with initial states  
   $\triangleright n$ , the number of iterations to run the simulation for  
  
   $t \leftarrow 0$   
  while  $t < n$  do  
     $\triangleright$  Change each entity's attributes according to its rule  
    for  $e \in E_t$  do  
       $E_{t+1} \leftarrow E_{t+1} \cup \{r_e(e)\}$   
    end for  
     $\triangleright$  Trigger events  
    for  $e \in E_t, (e, f, t, e_c) \in L_{\mathcal{W}}$  do  
       $\triangleright$  The event is triggered if the entity's attribute's value changed  
      if  $e(f) \neq e'(f), e' \in E_{t-1}$  then  
         $E_{t+1} \leftarrow E_{t+1} \setminus \{e_c\}$   
         $E_{t+1} \leftarrow E_{t+1} \cup \tau(e_c, t, \Delta_{\mathcal{F}}(f, e(f), e'(f)))$   
      end if  
    end for  
     $\triangleright$  Queue signals  
    for  $e \in E_t$  do  
       $S_t \leftarrow \eta_e(e)$   
    end for  
     $\triangleright$  Add signals as observations to entities that match modality  
    for  $s \in S_t, e \in E_t$  do  
      if  $m_s \in \mathcal{M}_e$  then  
         $O_e \leftarrow O_e \cup \{s\}$   
      end if  
    end for  
     $\triangleright$  Queue actions for entities with agents  
     $A_t = \{A_t^{e_1}, \dots, A_t^{e_{|E_t|}}\} \leftarrow \{\emptyset, \dots, \emptyset\}$   
    for  $e \in E_t$  do  
       $a \leftarrow \gamma(e)$   
      if  $a \neq 0$  then  
         $A_t^e \leftarrow A_t^e \cup \{\text{ACT}_a(M_e, O_e, \Sigma_e)\}$   
      end if  
    end for  
     $\triangleright$  Execute actions  
    for  $e_\epsilon \in E_t, a \in A_t^{e_\epsilon}$  do  
       $E_{t+1} \leftarrow E_{t+1} \cup \alpha_\epsilon(e_\epsilon, a)$   
    end for  
     $t \leftarrow t + 1$   
  end while  
end procedure
```

3 Mechanisms

These are descriptions of the ACT algorithm introduced in Section 2.3. All algorithms return a set of motor signals $a = (\alpha, v)$, $\alpha \in \mathcal{A}$, $v \in D_\alpha$. On the first iteration, at t_0 , the algorithms receive initial parameters. Then, parameters for the t -th iteration of the algorithm should be set in a previous iteration of that algorithm.

All algorithms for an entity have as parameters at least:

- the motor signal symbols and domains, \mathcal{A} and $D_{\mathcal{A}}$, or the set of possible motor signals derived from it A , where $a = (\alpha, v)$, $a \in A$, $\alpha \in \mathcal{A}$, $v \in D_\alpha$
- the observations at time t , O_t

3.1 Operant Conditioning

In the operant conditioning mechanism, behavior is reinforced by the presence of a rewarding stimulus. When a certain motor signal is followed by an observation that is ‘rewarding,’ then the probability of sending that motor signal will be increased. Conversely, when a motor signal is not followed by such a reward, this probability is decreased.

To produce this behavior, the algorithm maintains a probability distribution over all possible combinations of motor signals, $P : \Sigma \rightarrow [0, 1]$, selects motor signals by sampling from this distribution, and updates the distribution based on the experience resulting from the previously selected motor signal. For the babybot, there are four limbs that each have three possible directions of movement, resulting in probability distribution over $3^4 = 81$ possible combinations of motor signals.

The algorithm starts with the uniform distribution over these combinations. Every iteration of the algorithm, the probability of the combination of motor signals that was selected in the previous, $t - 1$ -th, iteration σ_{t-1} is updated dependent on the current observation O_t . If the observation contains at least one of a set of ‘rewards’ $R \subset O$, then the particular probability is increased by a certain amount δ_+ . Otherwise, the probability is decreased by a certain amount δ_- instead. After changing the probability, all probabilities in the distribution are renormalized to sum up to a probability of 1 again. The set of rewards that is used for the babybot consists only of the (*movement = faster*) observation. Fictional examples of the initial distribution and a possible distribution after an update can be found in Figure 5 and Figure 6.

The motor signal at iteration t is selected by sampling from the distribution. This means that any combination of motor signals is selected with the probability that is stored with it. For example, an combination of motor signals with a its probability set to $\frac{1}{81}$ will be selected with probability $\frac{1}{81}$.

Probabilities are forced to a minimum value p_{\min} during the decreasing step of the algorithm for two reasons. First, the minimum value prevents the algorithm from excluding a combination of motor signals for selection in future iterations. If at one point a probability is set to 0, then that probability will remain 0 for the subsequent iterations. A motor signal with probability 0 will not be selected, as the probability for selection would be 0, and consequently is not updated. The probability will also not change during normalization. As a consequence, once a particular motor signal’s probability reaches 0, changes in

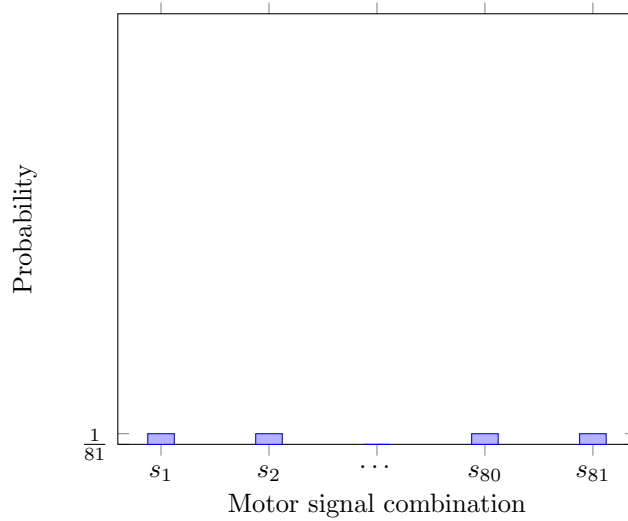


Figure 5: An example of an initial, uniform distribution for the babybot. There are $3^4 = 81$ possible combinations, so all combinations start with a probability of $\frac{1}{81}$ of being selected.

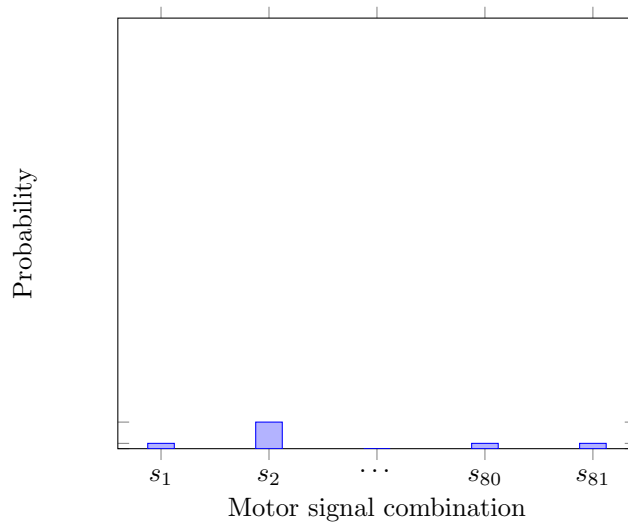


Figure 6: After a reward for combination s_2 , the probability for s_2 is updated and the distribution is normalized, resulting in lower probabilities for all other combinations.

the environment that would cause that motor signal to be reinforced normally would instead not cause the motor signal to be reinforced. An example of such a change is one where the limb that is connected to the mobile is switched, for example from the right foot to the left hand. Second, a conceptual constraint requires probabilities to be positive values.

3.1.1 Pseudo Code of Operant Conditioning

Algorithm 2 Operant conditioning algorithm

```

procedure OPERANTCONDITIONING( $O_t, \Sigma, R, P_t, \sigma_{t-1}, \delta_+, \delta_-, p_{\min}$ )
  ▷  $O_t$ , the observed variables and their values
  ▷  $\Sigma$ , the possible combinations of motor signals
  ▷  $R$ , the set of rewarding variable assignments
  ▷  $P_t$ , the current probability distribution
  ▷  $\sigma_{t-1}$ , previously selected combination of motor signals
  ▷  $\delta_+$ , amount to increase probability with
  ▷  $\delta_-$ , amount to decrease probability with
  ▷  $p_{\min}$ , minimum probability

   $P_{t+1} \leftarrow P_t$ 
  ▷ Determine whether a reward is observed
  if  $R \cap O_t \neq \emptyset$  then
    ▷ When the previously sent motor signal was followed by a reward,
    ▷ increase the probability for that motor signal
     $P_{t+1}(\sigma_{t-1}) \leftarrow P_t(\sigma_{t-1}) + \delta_+$ 
  else
    ▷ When the motor signal was not followed by a reward,
    ▷ decrease the probability for that motor signal
     $P_{t+1}(\sigma_{t-1}) \leftarrow \mathbf{max}(p_{\min}, P_t(\sigma_{t-1}) - \delta_-)$ 
  end if
  ▷ Normalize the probabilities
   $p_{\text{total}} \leftarrow \sum_{\sigma \in \Sigma} P_{t+1}(\sigma)$ 
  for  $\sigma \in \Sigma$  do
     $P_{t+1} \leftarrow \frac{P_{t+1}(\sigma)}{p_{\text{total}}}$ 
  end for

  ▷ Select a combination of motor signals by sampling from the probability
  distribution
  return SAMPLEWITHPROBABILITY( $\Sigma, P_{t+1}$ )
end procedure

```

3.2 Causal Learning

The causal learning mechanism learns the causal relations between variables from probabilistic data. New motor signals are selected as those motor signals that have the maximum likelihood of resulting in faster mobile movement, given the data.

By computing the (conditional) probabilities of variables from observations, a causal Bayesian network representing the causal relations of the data can be inferred from the conditional dependencies of variables through a constraint-based algorithm [7]. A Bayesian network represents the conditional dependencies of probabilistic variables as a directed acyclic graph; a causal Bayesian network represents causal relations between variables as a directed acyclic graph, where nodes represent variables and directed edges represent the causal relations between variables [8]. A causal Bayesian network differs from a Bayesian network in the way it is interpreted, i.e. causally. In case of the babybot, the variables that make up the probability distribution and thereby the causal Bayesian network, consist of the motor signal variables S , the limb position variables L , and the mobile movement variables M . and motor signals, where each has two instances to denote a time relation, as a ‘previous’ instance denoted by $\text{PREV}(\cdot)$ and as a ‘current’ instance denoted by $\text{CURR}(\cdot)$.

The constraint-based algorithm is a computationally intensive algorithm, as (conditional) dependencies between all variables are considered, resulting in exponential time complexity in the number of variables. Additionally there was difficulty of updating the network with new data points, as a new data point could entail change in any previously found (conditional) dependency. This demands the necessity to recalculate the whole network at every iteration, which, as mentioned before, is computationally intensive. This led us to consider whether the constraint-based algorithm is necessary to represent the causal relations in the data. Since the probability distribution over all the variables, $P(L, M, S)$, already encodes the causal relations, this probability distribution can be used for computations concerning these causal relations without the need of the constraint-based algorithm. The probability distributions used for the constraint-based algorithm and the method without the constraint-based algorithm are the same, although the latter does not have an explicit model of the causal relations in that it does not explicitly store the causal relations between variables, but that the causal relations are inferred from the (conditional) independencies in the observed probability distribution over all variables $P(L, M, S)$. Variables that are (conditionally) dependent of one another are interpreted as being causally related to each other. In this, the mechanism has a causal representation of the world, distinguishing it from the operant conditioning mechanism that does not have such a causal representation.

The algorithm starts an exploration phase that consists of ‘motor babbling’ to collect data. Then, the probability of mobile movement given the limb positions and motor signals is computed. Because the full probability distribution would be too large, the distribution is factorized into two tables to reduce the computational complexity. In assuming that the limbs’ positions are conditionally dependent on the motor signals, the full probability distribution can be factorized using Bayes’ rule: $P(A|B) = \frac{P(A)P(B|A)}{P(B)}$. Let L be all the limb position variables, M be the mobile movement variables, and S be the motor signal variables. Then the conditional probability of interest is the probability of mobile movement given the other variables: $P(M | L, S)$. After factorization,

this becomes:

$$P(M | L, S) = \sum_{l \in L} P(M | L = l) \cdot P(L = l | S)$$

$P(L | S)$ is the distribution over the limb positions, given the motor signals. This is a probability distribution of sixteen variables, i.e. the previous and current four limb positions, $2 \cdot 4$ variables, and the previous and current four motor signals, $2 \cdot 4$ variables, resulting in $2 \cdot 4 + 2 \cdot 4 = 16$ variables. Each variable has three possible values, resulting in a table with $3^{16} = 43105770$ probabilities. Potentially, the limb positions are dependent on any combination of motor signals. That is why the entire conditional probability table should be considered.

$P(M | L)$ is the distribution over the mobile movement variables given the limb position variables. This is a probability distribution over ten variables, i.e. the previous and current four limb positions, $2 \cdot 4$ variables, and the previous and current mobile movement, $2 \cdot 1$ variables, resulting in $2 \cdot 4 + 2 \cdot 1 = 10$ variables. Each variable has three possible values, resulting in a table with $3^{10} = 59049$.

This factorization reduces the number of entries that make up the probability table. In this case, the factorization into these two probabilities reduces the number of entries from a table of 18 variables, meaning 3^{18} entries, to two tables of individually 3^{16} and 3^{10} entries. This is a reduction by a factor of $\frac{3^{18}}{3^{16}+3^{10}} \approx 9$. This reduces the amount of data that is necessary to be able to learn the causal relations between the variables. However, regardless of this slight reduction, the probability table is very large, still requiring a large amount of data before probabilities can be computed that approximate the real probabilities.

New motor signals are selected from the conditional probability distribution by selecting the set of motor signals with the maximum likelihood of resulting in faster mobile movement. In calculating the maximum likelihood, values of the limb positions are marginalized over. The variables are assigned the values that are appropriate at the current iteration by retrieving them from memory, i.e. the previous motor signals, and mobile movement, and the previous and current limb positions, then the current mobile movement value is set to ‘faster’ and the set of current motor signals is selected that results in the highest conditional probability over this complete set of variables.

An example of the kind of structure that this mechanism learns is indicated below. Consider a babybot with just one arm with a position $\mathcal{F} = \{ap\}$, and motor signal $\mathcal{A} = \{a\}$, which is connected to a mobile, as described before where m is the mobile’s movement signal. Initially, the probability distribution is uniform, meaning all variables are independent, depicted as a causal Bayes net in Figure 7. For a visual representation of the possible dependencies in the Bayesian network for the babybot, see Figure 9

After some iterations, the infant could learn a causal relation as depicted in Figure 8.

3.2.1 Pseudo Code of Causal Learning

The set of motor signals A and observations O are combined into a single set of variables that will be used in the probability distributions. The variable symbols $\mathcal{V}' = \{v_1, \dots, v_{|\mathcal{V}'|}\}$. and domains $D = \{D_1, \dots, D_{|\mathcal{V}'|}\}$ of values D_i of $v_i \in \mathcal{V}'$.

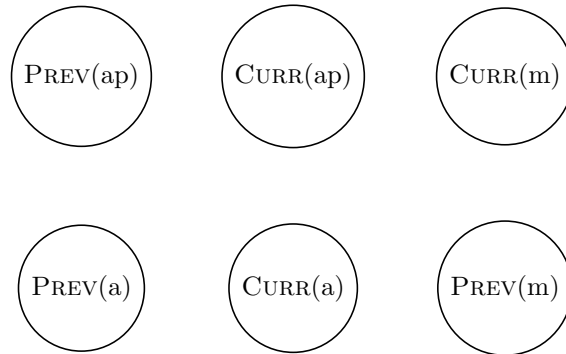


Figure 7: A Bayes net representing the underlying structure of an initial uniform probability distribution of a babybot. All variables are independent of each other. ap is the arm's position. a is the arm's motor signal. m is the mobile's movement.

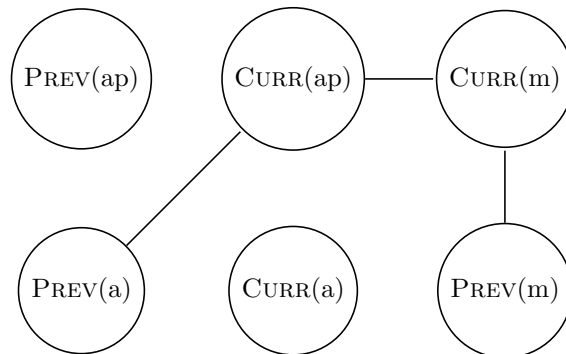


Figure 8: A fictional example Bayes net representing the underlying structure of an initial uniform probability distribution of a babybot. A causal relation between the previous arm motor signal and the current arm position, and the current arm position position and the current mobile's movement is learned. Edges are undirected because the direction of the causality cannot be inferred directly from the probability distribution. ap is the arm's position. a is the arm's motor signal. m is the mobile's movement.

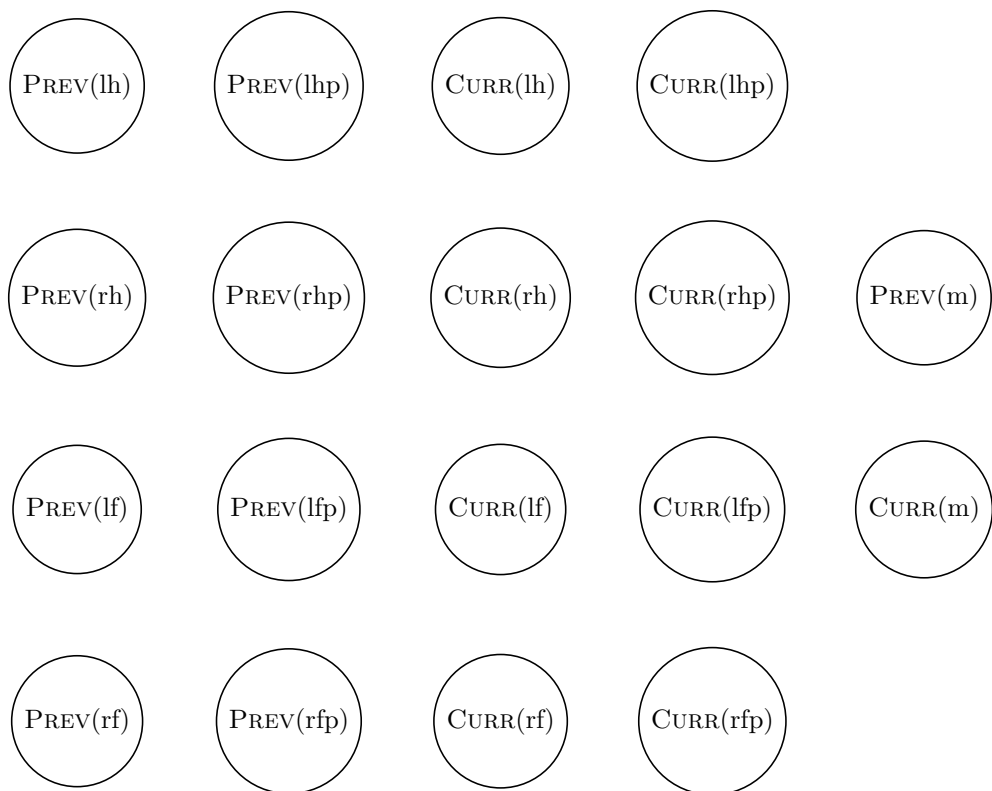


Figure 9: Bayesian network for the babybot. Given the assumptions, a variable can be dependent on variables on the left, but not on variables on the right side of that variable.

The motor signal symbols $\mathcal{V}'_M \subseteq \mathcal{V}'$, $\mathcal{V}'_M = \mathcal{A}$ and the sensory, i.e. non-motor signal symbols, $\mathcal{V}'_S \subseteq \mathcal{V}'$, $\mathcal{V}'_S = \mathcal{O}$ and the motor signal symbols and the sensory symbols are unique $\mathcal{V}'_M \cap \mathcal{V}'_S = \emptyset$.

These variable symbols are altered to represent a relation in time between variables by adding a ‘current’ or ‘previous’ denominator. This results in the variable symbols \mathcal{V} where $v \in \mathcal{V}' \rightarrow \text{PREV}(v) \in \mathcal{V}$, $\text{CURR}(v) \in \mathcal{V}$ with domains $\text{PREV}(v) \in \mathcal{V} \rightarrow D_{\text{PREV}(v)} = D_v$ and $\text{CURR}(v) \in \mathcal{V} \rightarrow D_{\text{CURR}(v)} = D_v$.

An assignment of a value to a variable $(v_i, d_{i,j}) \in (\mathcal{V} \times D_i)$ is denoted as $v_i = d_{i,j}$, and the set of all possible variable assignments is V .

The memory $M = \{(t_1, c_1), \dots, (t_{|M|}, c_{|M|})\}$ from the set of all possible memories \mathbf{M} , is the set of ordered pairs of time t_i and the assignment of values over variables c_i at time t_i . The notation M^i denotes the conjunction c_i at time t_i .

The point in time $t \in \mathbb{N}_0$ is a number that is used within the algorithm to store new observations in memory, implicitly storing time information as t is incremented every iteration.

P is a probability distribution over the observed variables, $P : (v \times D_v)^{\mathbb{N}} \rightarrow \mathbb{R}$. A probability over variables with specified values is denoted as $P(v_i = d_{i,j}, \dots, v_n = d_{n,k})$, a conditional probability is denoted as $P(v_i = d_{i,h}, \dots, v_j = d_{j,k} \mid v_{j+1} = d_{j,p}, \dots, v_n = d_{n,k})$ and the marginalized probability is denoted with a variable name without value $P(v)$.

The state $s \in S = \{s_{\text{explore}}, s_{\text{experiment}}\}$ is used as an indicator of the stage of the algorithm that we are in, as different things are calculated at different times t .

`SELECTRANDOMMOTOR SIGNALS` : $\Sigma^n \rightarrow \Sigma$ is a function that selects a combination of motor signals at random. This is only used in the exploration phase of the algorithm to gather an initial set of data. `CONDITIONALPROBABILITIES` : $\mathbf{M} \times \mathcal{V} \times \mathcal{V} \rightarrow P$ is a function that computes the conditional probability distribution $P(A|B)$ given a memory, the variables $B \subset \mathcal{V}$ that are conditioned on and the other variables $A \subset \mathcal{V}$. In the algorithm, this function is used to compute the probability distribution from the observations that are in memory up to that point in time.

Algorithm 3 Causal Learning Algorithm

procedure CAUSALLEARNING($t, O_t, \Sigma, \mathcal{V}'_M, \mathcal{V}'_S, M_t, s_t, n_{\text{explore}}, P_{\text{motor}}, (v_r, d_r)$)

- ▷ t , current iteration
- ▷ O_t , current observations
- ▷ Σ , all possible motor signals, i.e. combinations of motor signal symbols and values
- ▷ $\mathcal{V}'_M \cup \mathcal{V}'_S$, all variable symbols, divided into motor symbols and sensory symbols respectively
- ▷ M_t , the memory so far, a set of previous experiences
- ▷ n_{explore} , number of iterations in the exploration phase
- ▷ P_{motor} , conditional probability distribution calculated from exploration
- ▷ (v_r, d_r) , variable and value used for maximizing calculation

$\mathcal{V}_M = \{\text{PREV}(v) \mid v \in \mathcal{V}'_M\} \cup \{\text{CURR}(v) \mid v \in \mathcal{V}'_M\}$
 $\mathcal{V}_S = \{\text{PREV}(v) \mid v \in \mathcal{V}'_S\} \cup \{\text{CURR}(v) \mid v \in \mathcal{V}'_S\}$

if $t > n_{\text{explore}}$ **then**

- ▷ Change state, end the exploration phase
- $s_{t+1} \leftarrow s_{\text{experiment}}$
- ▷ Compute the conditional probability distribution, without mobile movement
- $\mathcal{V}^- \leftarrow \mathcal{V}_S \setminus \{\text{PREV}(v_r), \text{CURR}(v_r)\}$
- $P_{\text{motor}} \leftarrow \text{CONDITIONALPROBABILITIES}(M_t, \mathcal{V}^-, \mathcal{V}_M)$
- ▷ Clear memory, but keep a minimum to have data for the other conditional probability table
- $M_t \leftarrow M_t \setminus M_{t-3}$

end if

if $s_t = s_{\text{explore}}$ **then**

- ▷ Explore by random selection of actions
- $\sigma^{\text{sel}} \leftarrow \text{SELECTRANDOMMOTOR SIGNALS}(\Sigma)$

else if $s_t = s_{\text{experiment}}$ **then**

- ▷ Select the combination of motor signals with the highest probability of resulting in (v_r, d_r)
- ▷ Calculate the conditional probability distribution
- $P_{\text{limb}} \leftarrow \text{CONDITIONALPROBABILITIES}(M_t, \{v_r\}, \mathcal{V}_S \setminus \{v_r\})$
- $\mathcal{V}'_S^- \leftarrow (\mathcal{V}'_S \setminus \{v_r\})$
- ▷ Compute for the predefined variable for the current time, and take the stored value for the previous time
- $X_A \leftarrow \{(\text{CURR}(v_r), d_r), (\text{PREV}(v_r), M_t^{t-1}(v_r))\}$
- ▷ Compute for the stored values for the previous time, and sum over those for the current time
- $X_B \leftarrow \{\text{CURR}(v) \mid v \in \mathcal{V}'_S^-\} \cup \{(\text{PREV}(v'), M_t^{t-1}(v')) \mid v' \in \mathcal{V}'_S^-\}$
- ▷ Compute for the stored values of previous motor signal selections
- $X_C \leftarrow \{(\text{PREV}(v), M_t^{t-2}(v)) \mid v \in \mathcal{V}'_M\}$
- $\sigma^{\text{sel}} \leftarrow \arg \max_{\sigma^4 \in \Sigma^4} P_{\text{limb}}(X_A \mid X_B) \cdot P_{\text{motor}}(X_B \mid X_C \cup \{(\text{CURR}(v), d) \mid (v, d) \in \sigma^4\})$

end if

- ▷ Add the new experience to memory
- $M_{t+1} \leftarrow M_t$
- ADDEXPERIENCE($M_{t+1}, O_t, \sigma^{\text{sel}}$)
- return** σ^{sel}

end procedure

4 Results

Simulations were run with both the operant conditioning and causal learning mechanisms. Simulations ran for 300 iterations each and changes in limb position were recorded for every limb and grouped into 50 blocks of six iterations each. For every group, the resulting patterns of ten simulations were averaged to control for random effects. A size of ten was chosen as a compromise between number of simulations and simulation time, as the causal learning simulations took in the range of twenty minutes to an hour to simulate in a python simulation on a 3 GHz processor. The results are presented in the following plots.

4.1 Normal Condition

To compare how the mechanisms' pattern of behavior changes in an unchanging environment, both mechanisms were simulated in a condition in which the mobile's movement is controlled by movement of the right foot.

Figure 10: Operant conditioning mechanism. The mobile was connected to the right foot. Results averaged over 10 runs.

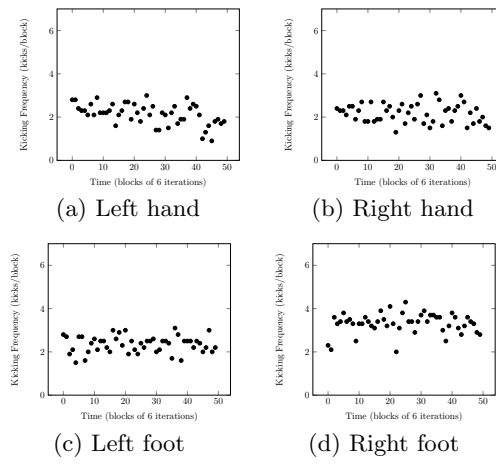


Figure 11: Causal learning mechanism. The mobile was connected to the right foot. Results averaged over 10 runs.

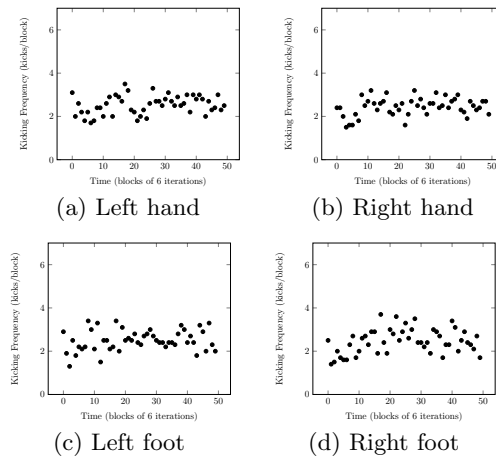


Figure 12: Operant conditioning mechanism, with only single limb movements. The mobile was connected to the right foot. Results averaged over 10 runs.

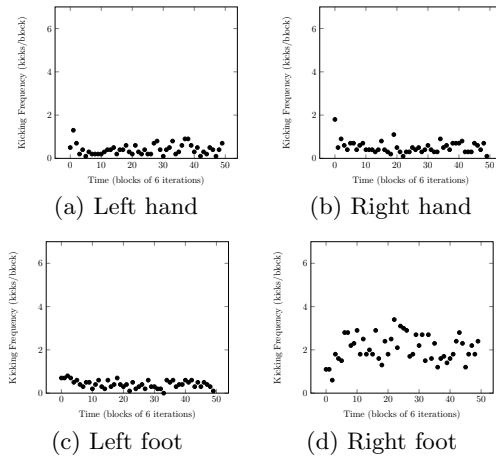
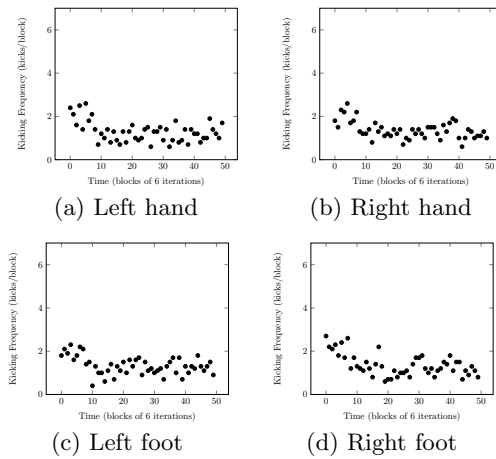


Figure 13: Causal learning mechanism, selecting for single limb movements. The mobile was connected to the right foot. Results averaged over 10 runs.



In the operant conditioning mechanism, Figure 10, the frequency of movement of the limbs does not go to zero. This could be because the distribution over the combinations of motor signals is approximately uniform because of the large number of combinations, causing the algorithm to select motor signals approximately at random if there is no combination of motor signals that was reinforced in the previous iteration. The frequency of movement of the right foot ($m_{rf} = 3.31$, $s_{rf} = 0.46$) is higher than that of other limbs ($m_{lh} = 2.14$, $s_{lh} = 0.48$, $m_{lf} = 2.35$, $s_{lf} = 0.38$, $m_{rh} = 2.18$, $s_{rh} = 0.44$). This is similar to the empirical results, where the frequency of movement of the connected limb is higher than the frequency of movement for the other limbs [5].

In the causal learning mechanism Figure 11, the limbs move with the same frequency ($m_{lh} = 2.53$, $s_{lh} = 0.43$, $m_{lf} = 2.50$, $s_{lf} = 0.47$, $m_{rf} = 2.46$, $s_{rf} = 0.56$, $m_{rh} = 2.44$, $s_{rh} = 0.41$). This indicates that the causal learning mechanism has not learned the causal relation between the right foot movement and the movement of the mobile. A possible explanation for this is that there was not enough data to learn this causal relation. This is not similar to the empirical results, as the empirical results showed an increase in the frequency of movement for the limb that was connected to the mobile.

In the operant conditioning mechanism where movement is constrained to a single limb, Figure 12, frequency of movement of the right foot ($m_{rf} = 2.06$, $s_{rf} = 0.61$) is higher than that of other limbs ($m_{lh} = 0.41$, $s_{lh} = 0.25$, $m_{lf} = 0.41$, $s_{lf} = 0.18$, $m_{rh} = 0.52$, $s_{rh} = 0.28$). Other limbs' frequencies, i.e. not the right foot, are near zero from the start; which is different from the empirical data where all limbs have an initial period of high frequency movement. The global pattern is similar to the pattern in Figure 10. This is again similar to the empirical results, although the low frequency of the limbs that are not connected to the mobile is lower.

In the causal learning mechanism where motor signals are constrained to single limb movement, Figure 13, the limbs move with the same frequency, as was also observed in Figure 11, $m_{lh} = 1.30$, $s_{lh} = 0.48$, $m_{lf} = 1.33$, $s_{lf} = 0.41$, $m_{rf} = 1.37$, $s_{rf} = 0.51$, $m_{rh} = 1.37$, $s_{rh} = 0.41$. The initial period of higher frequency movement indicates the exploration phase, after which the selected motor signals are selected from the previous data resulting in a drop in movement frequency. A possible explanation of why this pattern is different from the condition in which movement with any number of limbs is selected, is that the reduction in the number of possible combinations of motor signals causes the probability table to be filled faster. This would cause motor signals to be selected less frequently if there has already been an experience where almost all variables had the same value, as this would skew the conditional probability toward selecting the motor signal for that experience.

4.2 Switch Condition

To compare how the mechanisms adapt to changes in the environment, both mechanisms were also simulated in a condition where the limb that controls the mobile's movement is changed from the right foot to the left hand at the midpoint of the simulation.

Figure 14: Operant conditioning mechanism. The mobile was connected to the right foot, then switched to be connected to the left hand (indicated by the vertical line). Results averaged over 10 runs.

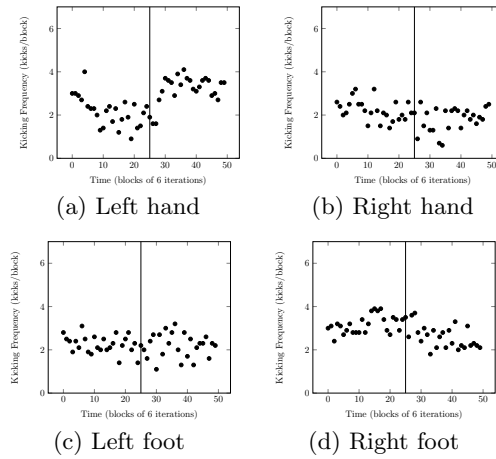


Figure 15: Causal learning mechanism. The mobile was connected to the right foot, then switched to the left hand (indicated by the vertical line). Results averaged over 10 runs.

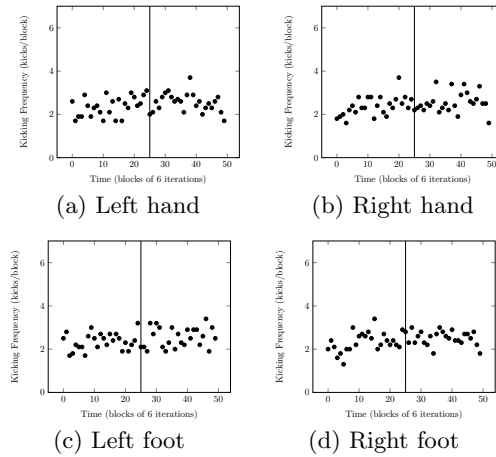


Figure 16: Operant conditioning mechanism. Only one limb is moved at the same time. The mobile was connected to the right foot at first, then switched to the left hand (indicated by the vertical line). Results averaged over 10 runs.

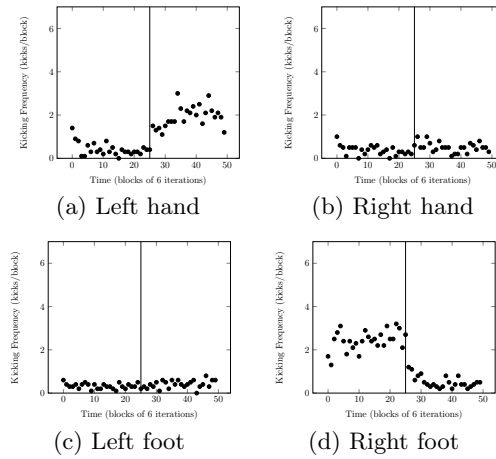
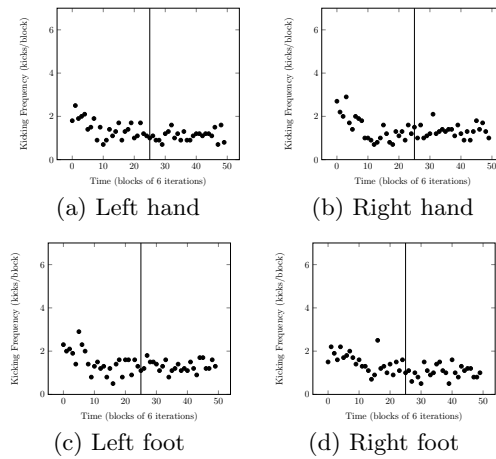


Figure 17: Causal learning mechanism, selecting for single limb movements. The mobile was connected to the right foot, then switched to be connected to the left hand (indicated by the vertical line). Results averaged over 10 runs.



In the operant conditioning mechanism, Figure 14, before the switch the frequency of movement of the right foot ($m_{rf} = 3.16$, $s_{rf} = 0.40$) is higher than that of other limbs ($m_{lh} = 2.17$, $s_{lh} = 0.68$, $m_{lf} = 2.26$, $s_{lf} = 0.41$, $m_{rh} = 2.22$, $s_{rh} = 0.48$). After switching, the left hand's frequency of movement ($m_{lh} = 3.18$, $s_{lh} = 0.65$) is higher than before switching ($m_{lh} = 3.18 > m_{lh} = 2.17$) and it is higher than the frequencies of other limbs after switching ($m_{lf} = 2.20$, $s_{lf} = 0.54$, $m_{rf} = 2.61$, $s_{rf} = 0.52$, $m_{rh} = 1.81$, $s_{rh} = 0.54$). This indicates that before switching motor signals of right foot movement are reinforced, whereas after switching the motor signals of left hand movement are reinforced, causing the previously reinforced right foot movement's probabilities to decrease.

In the causal learning mechanism, Figure 15, all frequencies are similar. Before switch: $m_{lh} = 2.36$, $m_{lf} = 2.35$, $m_{rf} = 2.32$, $m_{rh} = 2.38$, $s_{lh} = 0.44$, $s_{lf} = 0.38$, $s_{rf} = 0.45$, $s_{rh} = 0.45$. After switch: $m_{lh} = 2.54$, $m_{lf} = 2.54$, $m_{rf} = 2.52$, $m_{rh} = 2.56$, $s_{lh} = 0.42$, $s_{lf} = 0.45$, $s_{rf} = 0.32$, $s_{rh} = 0.46$. This is again an indication that the causal relations are not learned.

In the operant conditioning mechanism selecting for single limb movements, Figure 16, a pattern similar to the one in Figure 14 is observed. Before switching, the frequency of movement of the right foot ($m_{rf} = 2.42$, $s_{rf} = 0.46$) is higher than that of other limbs ($m_{lh} = 0.42$, $s_{lh} = 0.30$, $m_{lf} = 0.33$, $s_{lf} = 0.12$, $m_{rh} = 0.22$, $s_{rh} = 0.22$). After switching, the left hand's frequency of movement ($m_{lh} = 1.86$, $s_{lh} = 0.56$) is higher than before switching ($m_{lh} = 1.86 > m_{lh} = 0.42$) and higher than that of other limbs ($m_{lf} = 0.40$, $s_{lf} = 0.18$, $m_{rf} = 0.60$, $s_{rf} = 0.50$, $m_{rh} = 0.51$, $s_{rh} = 0.23$).

In the causal learning mechanism selecting for single limb movement, Figure 17, a pattern similar to that in Figure 13 is observed. Limbs' movement frequencies are similar before switching: $m_{lh} = 1.44$, $m_{lf} = 1.51$, $m_{rf} = 1.50$, $m_{rh} = 1.43$, $s_{lh} = 0.43$, $s_{lf} = 0.55$, $s_{rf} = 0.44$, $s_{rh} = 0.60$. And after switching $m_{lh} = 1.10$, $m_{lf} = 1.30$, $m_{rf} = 1.03$, $m_{rh} = 1.32$, $s_{lh} = 0.24$, $s_{lf} = 0.25$, $s_{rf} = 0.29$, $s_{rh} = 0.28$. As in the normal condition, the initial period of higher frequency movement indicates the exploration phase, after which the selected motor signals are selected from the previous data resulting in a drop in movement frequency.

5 Discussion

To gain evidence for whether the experiment by Rovee-Collier et al. can be used to determine presence of a sense of agency, a comparison was made between the behavioral patterns of an operant conditioning and a causal learning mechanism in relation to the empirical data. These two mechanisms were selected because one fulfilled no necessary conditions for a sense of agency while the other does fulfil such a condition. The comparison's results implied that data such as in [5] cannot be used to conclude a sense of agency in infants.

Movement patterns resulting from the operant conditioning mechanism are similar to movement patterns resulting from the infants in the experiment in that the frequency of movement of the limb that is connected to the mobile is higher than the frequency of movement of the other limbs. A difference between the two mechanisms and the infants is that in the patterns resulting from the infants, the movement frequency of the limbs that were not connected to the mobile decreased. In the operant conditioning mechanism, this decrease in frequency is absent. This could indicate that the operant conditioning mechanism could be changed to result in more similar movement patterns. One such change could be a different, more sophisticated reinforcement scheme that reinforces individual motor signals instead of combinations of motor signals. One explanation for the operant conditioning mechanism's results is that the reduction of the complex world into the presence or absence of a reward, essentially a binary value, greatly reduces the complexity of the learning problem.

Movement patterns resulting from the causal learning mechanism are not similar to the movement patterns resulting from the infants. There is no shift to relative high frequency movement of the connected limb. This is consistent with the observation that there is no shift after changing the connected limb, which could indicate that more time is needed to learn the causal relations. One explanation for the causal learning mechanism's results is that the world model's complexity is too high: the number of experiences necessary to fill the probability table is exponential in the number of variables, which is intractable even for simple worlds. For the babybot in this thesis, the probability distribution contained eighteen variables, requiring in the order of 3^{18} experiences to completely fill the table. This is clearly much more than the number of simulated iterations. However, this mechanism should not be dismissed completely, as an infant might collect enough data during infancy to have a meaningful probability distribution. An exploration phase has been included in the algorithm, but it might have been too short. Future research could include a separate, extended exploration phase that learns the causal representation of the 'body' before entering the experiment environment. Infants might also constrain the causal model further, for example by only allowing a particular limb's motor signal variable to be connected to that same limb's position variable.

From these two observations, it follows that the operant conditioning mechanism which does not have a representation of the relation between the actions and the environment results in pattern similar to the infants' movement patterns. Conversely, the causal learning mechanism which does have such a representation results in a different pattern. Thus, this implies that data such as in [5] cannot be used to conclude a sense of agency in infants.

By designing a simulation of the mechanisms behind infant behavior, insight was gained into the problems that appear if infants actually implement these

mechanisms. This can be used in the generation of hypotheses, and the resulting mechanisms can be used to test hypotheses in developmental research. In turn, the knowledge that results from this can be used in developmental research to explore and develop models, and in artificial intelligence to develop new techniques, including learning mechanisms.

5.1 Scientific Challenges

During the research project, we encountered several scientific challenges. One challenge was the dynamic updating of a causal Bayes net. In an earlier stage of describing the causal learning mechanism, the algorithm calculated a causal Bayes network from the gathered observations. Because no method for dynamically updating such a network was available and recalculating the network was too computationally intensive, part of this algorithm was abandoned. This is an indication that there might be benefits to research on dynamic updating of such models; in general and perhaps specifically for developmental science.

Another big challenge was the problem of having the mechanisms consider only the relevant models or relations, being reminiscent of the frame problem in first-order logic [9]. For example, for the causal learning mechanism, even in a seemingly simple world as the one that was presented in this project, the number of experiences that is necessary to have enough data to be able to predict causal relationships suffers from a learnability problem. In the context of developmental research and AGI, this could be considered as another piece of evidence for the idea that the complexity of human intelligence might not (only) be found in the adult stage of life, but also in the stages before that. This involvement of the frame problem suggests that artificial intelligence research might benefit from developmental research. By exploring solutions that actual humans use to solve these hard problems, we could advance further towards artificial human-level intelligence.

Some decisions were made to reduce the complexity of the real problem, both for feasibility and because of time constraints. First, even though the original problem's complexity is underestimated by simplifying the infant's body and senses, there is still a problem of determining how much of the world is relevant. Four 1-dimensional 'limbs' are not as complex as four limbs that have joints and that are controlled by muscles that are in turn controlled by electrical signals. Still then, the causal learning mechanism, when considering all possible combinations of variables, requires a lot of data to learn the causal relations. In further research, more complex models could be explored to approach the real complexity of the problem, which includes continuous models and models in 3-dimensional space. This would entail different control mechanisms, and different feedback that might include noise.

Second, related to the first point, no interactions between limbs have been modelled, whereas in real infants movement of the right leg might cause the left leg's position to change also. Future research could include simple probabilistic models of these interactions or models of the infant's body.

Third, the babybot is assumed to always observe the mobile. This is not realistic, as infants move their head, blink, and move their arms in front of their eyes. Further research could explore models that introduce noise in this observation in some way, or that include head movement into the variables that are to be learned.

Fourth, there is no feedback from pressure when a limb is moved against the crib. Including this and other kinds of feedback could speed up learning of the inability to move a limb in the ‘down’ position and perhaps the ‘up’ position.

Fifth, in the current model, there is no speed component to the babybot’s movements. Different limb movement speeds affect the mobile’s speed differently, which adds complexity to the learning problem. However, for the purposes of this experiment, i.e. regarding learning contingencies between actions and effects, a constant speed allowed for comparing the algorithms.

6 Conclusion

We investigated the differences between operant conditioning and causal learning mechanisms in how far these mechanisms can explain behavior in infants during a learning task. These two mechanisms were selected because one fulfilled no necessary conditions for a sense of agency while the other does fulfil such a condition. Data obtained from the simulation of these two mechanisms were compared to the empirical data. These data implied that data such as in [5] cannot be used to conclude a sense of agency in infants.

During the research project, we encountered several fundamental scientific challenges. This indicates that interdisciplinary research between developmental research and artificial intelligence research is beneficial for both fields.

References

- [1] Aaron Sloman. Why some machines may need qualia and how they can have them: Including a demanding new turing test for robot philosophers. In *AI and consciousness: Theoretical foundations and current approaches AAAI fall symposium*, pages 9–16, 2007.
- [2] Aaron Sloman and Ron Chrisley. Virtual machines and consciousness. *Journal of Consciousness Studies*, 10(4-5):133–172, 2003.
- [3] Sam Adams, Itmar Arel, Joscha Bach, Robert Coop, Rod Furlan, Ben Goertzel, J Storrs Hall, Alexei Samsonovich, Matthias Scheutz, Matthew Schlesinger, et al. Mapping the landscape of human-level artificial general intelligence. *AI Magazine*, 33(1):25–42, 2012.
- [4] Philippe Rochat and Tricia Striano. Perceived self in infancy. *Infant Behavior and Development*, 23(3):513–530, 2000.
- [5] Carolyn Kent Rovee-Collier, Barbara A Morrongiello, Mark Aron, and Janis Kupersmidt. Topographical response differentiation and reversal in 3-month-old infants. *Infant Behavior and Development*, 1:323–333, 1978.
- [6] Hama Watanabe and Gentaro Taga. Initial-state dependency of learning in young infants. *Human movement science*, 30(1):125–142, 2011.
- [7] Alison Gopnik, Clark Glymour, David M Sobel, Laura E Schulz, Tamar Kushnir, and David Danks. A theory of causal learning in children: Causal maps and Bayes nets. *Psychological review*, 111(1):3, 2004.
- [8] Björn Meder. Seeing versus doing: Causal bayes nets as psychological models of causal reasoning. *Unpublished doctoral dissertation, Universität Göttingen*, 2006.
- [9] Patrick J Hayes. *The Frame Problem and Related Problems on Artificial Intelligence*.