Equivalence Projective Simulation as a Framework for Modeling Formation of Stimulus Equivalence Classes

Asieh Abolpour Mofrad[1]
Anis Yazidi[1]
Hugo L. Hammer[1]
Erik Arntzen[2]
[1]Dept. of Computer Science, OsloMet - Oslo Metropolitan University, Oslo, Norway
[2]Dept. of Behavioral Science, OsloMet - Oslo Metropolitan University, Oslo, Norway

Keywords: Stimulus equivalence classes, projective simulation, equivalence projective simulation, reinforcement learning, connectionist models

Abstract

Stimulus Equivalence (SE) and Projective Simulation (PS) both study complex behavior; the former in human subjects and the latter in artificial agents. We apply PS learning framework for modeling the formation of equivalence classes. For this purpose, we first modify PS model to accommodate imitating the emergence of equivalence relations. Later, we formulate the SE formation through the Matching to Sample (MTS) procedure. The proposed version of PS model, called Equivalence Projective Simulation (EPS) model, is able to act within a varying action set and derive new relations without receiving feedback from environment. To the best of our knowledge, it is the first time that the field of equivalence theory in behavior analysis is linked to an artificial agent in machine learning context. This model has many advantages over the existing neural network models. Briefly, our EPS model is not a black-box model, but rather a model with the capability of easy interpretation and flexibility for further modifications. To validate the model, some experimental results performed by prominent behavior analysts are simulated. The results confirm that EPS model is able to reliably simulate and replicate the same behavior as real experiments in various settings including formation of equivalence relations in typical participants, non-formation of equivalence relations in language-disabled children, and nodal effect in a linear series with nodal distance five. Moreover, through a hypothetical experiment we discuss the possibility of applying EPS in further equivalence theory research.

1 Introduction

In this paper, we will present a novel machine learning model that is able to efficiently replicate human behavior in equivalence experiments. The main stream of research in modeling equivalence behavior for humans using connectionist models involve neural networks. Despite being far less complex than neural network based models, our model is easy to interpret and flexible enough to model a wide range of behaviors in a matching-to-sample (MTS) experiment.

Sidman introduced stimulus equivalence term (Sidman, 1971) which later (Sidman & Tailby, 1982) is characterized through mathematical relations in equivalence sets i.e. reflexivity, symmetry, and transitivity between members of an equivalence class. By training some relations in a class, experimenter could test the emergence of new relations or derived relations upon the trained relations. As a general rule, a class composed of n stimuli, needs only $(n \quad 1)$ stimulus-stimulus matches to be trained. Each component of these relations must be used in at least one trained relation, and further none of the trained relations can have the same two stimuli as components. Given these constraints, there exist many possible ways for selecting training relation sets, some of them might be more efficient than the others (Fields et al., 1990; O'Mara, 1991; Arntzen & Holth, 1997; Hove, 2003; Lyddy & Barnes-Holmes, 2007; Arntzen et al., 2010a; Arntzen & Hansen, 2011; Fienup et al., 2015).

Stimulus equivalence framework as a learning method was originally used to teach children and adults with developmental disabilities like Autism and Down's syndrome (Sidman et al., 1974; Groskreutz et al., 2010; Toussaint & Tiger, 2010; Arntzen et al., 2010b; McLay et al., 2013; Arntzen et al., 2014; Ortega & Lovett, 2018). However, the equivalence theory can be used in teaching new concepts to normal children and adults, including college students (Sidman et al., 1986; Hove, 2003; Saunders et al., 2005; Fienup et al., 2010; Walker et al., 2010; Lovett et al., 2011; Grisante et al., 2013; Placeres, 2014; Fienup et al., 2015). Some neurocognitive disorders, like Alzheimer's disease, are also a research area that equivalence theory deals with where it is discussed that derived relational responding is deteriorated as the cognitive impairment advances over time (Bódi et al., 2009; Gallagher & Keenan, 2009; Steingrimsdottir & Arntzen, 2011; Sidman, 2013; Arntzen et al., 2013; Arntzen & Steingrimsdottir, 2014; Seefeldt, 2015; Ducatti & Schmidt, 2016; Arntzen & Steingrimsdottir, 2017; Brogård-Antonsen & Arntzen, 2019).

One interesting feature of stimulus equivalence is its efficiency and the fact that just a small fraction of relations has to be explicitly taught. This could make a faster intervention in disorders. By training only on few relations, less training trials are needed as the rest of relations can be simply deduced.

The stimulus equivalence relationship to verbal behavior is another interesting research topic in equivalence literature. For instance in (Hall & Chase, 1991) it is discussed that all equivalence classes could be defined as verbal behavior, but all verbal behavior can not be fit into equivalence classes. Moreover, the evidence shows that stimulus equivalence relations are not formed properly in nonverbal humans (Devany et al., 1986) and animals (Nissen, 1951; Sidman et al., 1982; Hayes, 1989). Furthermore, the Relational Frame Theory (RFT) is a psychological theory of human language which is built upon equivalence theory (Hayes, 1991, 1994; Barnes-Holmes & Roche,

2001). Relational frame theory describes stimulus equivalence research in relation to Skinner's verbal behavior, and model the beginning feature of human cognition (see, e.g., Barnes, 1994; Clayton & Hayes, 1999; Barnes-Holmes et al., 2000; Hayes & Sanford, 2014; Hayes et al., 2017, for more details on RFT research).

Investigations in the area of stimulus equivalence, traditionally, have employed humans or animals as experimental participants. However, artificial neural network (ANN) models of cognition, often referred to as *connectionist models (CMs)* (see, e.g., McClelland et al., 1987; Bechtel & Abrahamsen, 1991; Commons et al., 2016, for CMs) have been developed to simulate the behavior of human participants in stimulus equivalence experiments. Connectionism tries to explain and replicate intellectual abilities using artificial neural networks (McClelland et al., 1987). Many researchers have been exploring methods in which artificial neural networks could develop the understanding of *derived stimulus relations*, by using simulated MTS procedures (Barnes & Hampson, 1993; Cullinan et al., 1994; Lyddy et al., 2001; Tovar & Westermann, 2017) or by training stimulus relations through compound stimuli and alternative procedures to MTS (Tovar & Chávez, 2012; Vernucio & Debert, 2016). A connectionist model of RFT is presented in (Barnes & Hampson, 1997).

Connectionism brings a common conceptual and empirical domain for both behavior analysis and cognitive science (Fodor & Pylyshyn, 1988; Staddon & Bueno, 1991; Barnes & Holmes, 1991; Barnes & Hampson, 1993). Developing connectionist models of equivalence formation could be a tool to study the limitations and power of connectionism. For instance, modeling formation of stimulus equivalence classes shows that semantic and syntactic relations can be modeled through connectionist networks (Barnes & Hampson, 1993) as opposed to discussion whithin (Fodor & Pylyshyn, 1988).

The development of computational models makes it possible to examine variables that are challenging to examine on humans or animals due to time constrains or ethical issues. For instance, components of the computational model can be easily manipulated, disrupted, impaired, and removed to see the effect of those components on the results. Having more control over the experimental variables including a controllable environment is a major advantage of these models over experiments with human and animal subjects (Barnes & Hampson, 1993; McClelland, 2009; Ninness et al., 2018).

Computational models could be used for exploring the implications of new ideas through simulation (McClelland, 2009). Behavior-analytic researchers can apply artificial neural networks to understand, simulate, and predict derived stimulus relations made by human participants. Furthermore, a good model of complex behaviors like formation of stimulus equivalence classes will lead to a better understanding of the disorders which applied behavior analysis deals with and might enable us to suggest new interventions for patients (Murre et al., 2001; Baddeley et al., 2003).

On the other hand, the experimental data from humans could enhance the model of brain function in an efficient way. Similar to studies with human subjects, patients' data is a valuable source to make the model more realistic. For instance, knowing that people with dementia might not be able to derive transitive relation (Arntzen et al., 2013), would be an aid to advance the model.

Although neural networks is one of the most powerful simulation techniques, its

black-box nature makes interpreting it hard (Zhang et al., 2018)[1] and there are serious discussions to design models that are inherently interpretable instead of black-box models (see Rudin, 2019, for instance).

Moreover, in general, the computational power comes from a complex network that replicates the complex behavior appropriately, but does not help to understand the underlying mechanisms of the brain (see, e.g., Silver et al., 2016, deep neural network model) and (see, e.g., Mnih et al., 2015, deep reinforcement learning model). Among different types of machine learning schemes, Reinforcement Learning (RL) (Sutton & Barto, 2018) is the closest computational model to actual learning in humans and other animals, and many RL algorithms are inspired by biological learning systems such as the stimulus-response theory from behavioral psychology.

The newly developed idea of projective simulation (PS) agents (Briegel & De las Cuevas, 2012) can be seen as a RL algorithm. Projective simulation (Briegel & De las Cuevas, 2012; Mautner et al., 2015) provides a flexible paradigm that can be easily extended, a feature that makes it a suitable framework for equivalence class formation. PS is not a black-box and although it is a fairly simple graphical model, we will demonstrate that it is powerful enough to model equivalence class formation.

We propose a modified version of PS in order to make the model appropriate for equivalence modeling. The modification of PS model, not only makes it suitable for producing equivalence emergence, but also adds extra features to PS model that can be further used in machine learning research. Indeed, by studying how the brain works in equivalence theory, we can devise more intelligent algorithms that mimic human nature and which can be applied in other fields.

The outline of the paper is as follows; in section 2, the required background from stimulus equivalence and projective simulation is provided. The state of the art computational models of equivalence formation is discussed and compared to the newly presented model. Moreover, PS is compared with standard reinforcement learning methods and the motivation behind choosing PS as the basis of our model is provided. In section 3, the modified version of PS, called EPS hereafter, is presented. Section 4 brings the artificial model results from EPS and compare it to the results of real experiments in order to demonstrate that the model can produce realistic results despite its simplicity. Finally, in section 5 concluding remarks and further suggestions are provided.

2 Background and Related Works

To address the required background of this work, first we explain the concept of stimulus equivalence and some methods that are used to learn and test the relations in behavior analysis in section 2.1. In section 2.2, some computational models and connectionist models of stimulus equivalence class formation are discussed. Then, in section 2.3, the projective simulation as a model of intelligence machines is explained. The standard reinforcement learning (RL) models are compared with PS in section 2.4 and the reasons behind selection of PS framework are discussed.

[1] We mean by black-box that; although we can get accurate predictions from the model, we cannot explain or identify the logic behind the predictions in a clear way.

2.1 Stimulus Equivalence

Stimulus equivalence research is about complex human behavior research, including research on memory and problem solving, that formerly was just studied by cognitive psychology (Sidman, 1990). The stimulus-equivalence methodology introduced by Sidman (Sidman, 1994) uses MTS procedures to train arbitrary relations between unfamiliar stimuli, and deals with testing some derived relations through reflexivity, symmetry, transitivity and equivalence[2].

The MTS or conditional discrimination procedure occurs when a stimulus, say A_1, is given, and it must be paired with B_1 among a set of comparison stimuli, say B_1, B_2, and B_3. The discrimination is done through feedback or rewards provided by experimenter, and not because of resemblance between the matched stimuli. This arbitrary match between stimuli, is the key aspect to study the emergence of equivalence relations that are not matched directly (Sidman, 2009).

Two main procedures in behavior analysis for training the relations are MTS which uses *simple* stimuli; (see, e.g., Sidman, 1971; McDonagh et al., 1984; Sidman et al., 1986; Arntzen, 2012), and the go/no-go procedure or successive matching-to-sample (S-MTS) that uses *compound* (or *complex*) stimuli; (see, e.g., Markham & Dougher, 1993; Debert et al., 2007, 2009; Grisante et al., 2013; Lantaya et al., 2018). In MTS, which is the traditional procedure, a sample stimulus will be paired with one of the given choices, whilst in compound stimuli a match is shown and the participant learns if it is a correct match or not through trial and error (see, e.g., Grisante et al., 2013; Lantaya et al., 2018, for comparison of the procedures).

In equivalence literature, three training structures have been used in establishing conditional discrimination with MTS procedure: linear series (LS), many-to-one (MTO), and one-to-many (OTM) (Arntzen, 2012). For instance, if any of equivalence classes have four members each from one of A, B, C, and D categories, the order of training relations would be: AB, BC, and CD in LS; AD, BD, and CD in MTO; and AB, AC, and AD in OTM settings. However, a mixture of these methods is also a possibility; like AB, BC, and DC.

Conditional discrimination procedures might also be either simultaneous MTS or delayed MTS. In simultaneous MTS, a sample stimulus is presented which might require response[3]. Subsequent to the response, the comparison stimuli will appear. Both sample and comparisons remain on the screen until one of the comparisons is selected. However, in delayed MTS, the sample stimulus appears and disappears first. Then, the comparison stimuli appears after a certain time delay which could be fixed, called fixed delayed MTS, or changing, called titrated delayed MTS.

[2] Arbitrary MTS means there is no conceptual relation between an equivalence class members.

[3] The standard MTS procedure requires that the sample stimulus receives response by the participant before the comparison stimuli appears (say by clicking on the sample stimulus in computer setting experiments or by touching it in a physical setting). This guarantees that the sample stimulus has been observed. Sometimes there is no need to response but a delay between appearance of sample stimuli and responses (usually 1 2 s) is considered.

The performance evaluation of participant is usually done according to the criterion that participant must pass in order to be considered as mastery in training phase. After mastery of the training relations, the testing phase will be done. Note that mastery criterion ratio should be placed higher in training (e.g., $0{:}95 \quad 1$) than in testing (e.g., $0{:}9 \quad 1$), and that in the testing phase there is no feedback from experimenter.

The equivalence class is considered to be formed whenever the evidence (passing the criterion for testing) shows that all these relations are established (Sidman & Tailby, 1982). For more details about MTS training and testing procedures and parameters in formation of stimulus equivalence classes see for instance (Arntzen, 2012).

2.2 Computational Models of Formation of Stimulus Equivalence Classes

There are two main families of equivalence simulation methods: the first family of methods simulate MTS procedures that consider simple stimuli (Barnes & Hampson, 1993; Cullinan et al., 1994; Lyddy et al., 2001; Tovar & Westermann, 2017) while the second family of methods simulate equivalence formation through compound stimuli (Tovar & Chávez, 2012; Vernucio & Debert, 2016).

One of the well known behavior-analytic approaches to neural network is RELNET; the network for relational responding, which is a feed-forward neural network with back propagation learning (Barnes & Hampson, 1993). The model consists of three modular stages, first stage is an *encoder* that preprocesses the stimuli for the second stage which is called *relational responding machine* (central system), and the third stage is a *decoder* that decodes the output of the relational responding machine. The three stages are separate modules, the encoder and decoder act like a simple pattern association whilst the simulation of learning task is done through the central system. RELNET simulates MTS procedure for training and testing trials of conditional relations. It is used (Barnes & Hampson, 1993) to replicate a contextual control of derived stimulus relations in a real experiment (Steele & Hayes, 1991), and to study the effect of training protocols in equivalence class formation (Lyddy & Barnes-Holmes, 2007) by modeling the experiment in (Arntzen & Holth, 1997). One of the critics to the RELNET model is that the transitive relations were partially trained during encoding and therefore not derived as it supposed to (Tovar & Chávez, 2012).

Another computational model that uses MTS procedure is presented in (Tovar & Westermann, 2017). The model is a fully interconnected neural network that links equivalence class field to Hebbian learning, associative learning and categorization. The model assumptions are threefold; first, each neuron accounts for a stimulus that represented through activation. Second, the weighted connections between different neurons spread activation in the network, and third, the coactivation of neurons based on Hebbian learning, updates the connection weights and as a result, the network learns the relatedness of relations; both trained and derived. The model simulates three high impact studies (Sidman & Tailby, 1982; Devany et al., 1986; Spencer & Chase, 1996) and the connection weights in the model were compared with the results of real experiments which validates the model in various scenarios, such as the replication of failures in transitive responding for the experiment with disabilities (Devany et al., 1986).

Another promising alternative to MTS is to train stimulus equivalence relations with compound stimuli procedures (Tovar & Chávez, 2012). The network input in this case is stimulus pairs (e.g., A_1B_1, A_1B_3) and the output is yes/no responses. This model requires a previous learning of all possible relations of an equivalence class, say XYZ, in order to be able to make derived relations in desired classes. A replication of (Tovar & Chávez, 2012) using a go/no-go procedure, i.e. just considering a yes responses, is presented in (Vernucio & Debert, 2016). Both connectionist models are capable of simulating humans emergence of derived stimulus relations without the assistance of sample marking duplicators, that RELNET needs. Although RELNET, go/no-go and yes/no models are promising models, they are criticized for their inability to describe relatedness between members of stimulus classes, and that they are not considered to be biologically plausible (Tovar & Westermann, 2017; O'Reilly & Munakata, 2000).

The neural network presented in (Lew & Zanutto, 2011) is a real time neurocomputational model based on biological mechanisms which is able to learn various tasks including operant conditioning and DMTS. The network has three layers; the first layer receives sensory input from the environment and produce *short-term memory traces* for them. The second layer allows further filtering of task relevant stimuli, which will then be associated with the proper response in the third layer (see Rapanelli et al., 2015, for an application of the model).

A good overview of existing CMs is provided in (Ninness et al., 2018) along with a working example of a neural network called emergent virtual analytics (EVA) (see Ninness et al., 2019, for more simulations with EVA). Through EVA, the process of applying neural network simulations in behavior-analytic research is demonstrated.

As aforementioned, in the current study, we model the MTS procedure and use simple stimuli based on PS as a machine learning framework. The proposed model is not a connectionist model, but a reinforcement learning agent, that is biologically plausible and uses Hebbian learning principles.

2.3 Projective Simulation

Projective Simulation, introduced recently (Briegel & De las Cuevas, 2012), is a machine learning models built upon principles from physics which relies on stochastic processing of experience. PS model can be seen as a reinforcement learning algorithm that can be embodied in an environment, perceive stimuli, execute actions, and learn through trial and error.

PS has a neural network type structure that is considered to be its physical basis, where any initial experience can activate other patterns in a spatio-temporal manner. The memory type in PS denoted as episodic[4] & compositional memory (ECM), which literally is a directed, weighted network of clips, where each clip represents a remembered percept, action, or sequences of them. Episodic & compositional memory can be described as a probabilistic network of clips. In the following, we use the terms episode and clip interchangeably.

[4]Episodic memory introduced in psychology in the 1970s by Tulving and Ingvar and it has gained increasing attention in the cognitive neuroscience and in other scientific fields.

Once a percept is observed, its coupled clip is activated and a random walk on the clip network triggers, until an action clip is reached and coupled out as a real action that agent does. In other words, any recall of memory is understood as a dynamic replay of an excitation pattern, which gives rise to episodic sequences of memory, see Figure 1 for a demonstration.

Indeed, in PS model, a random walk in the network of clips happens before the action is excited. An interpretation is that the agent *projects* itself to the future (think of what will happen if an action is chosen) and therefore complex decisions might be taken, including choices that were not in the training phase (like stimulus equivalence).

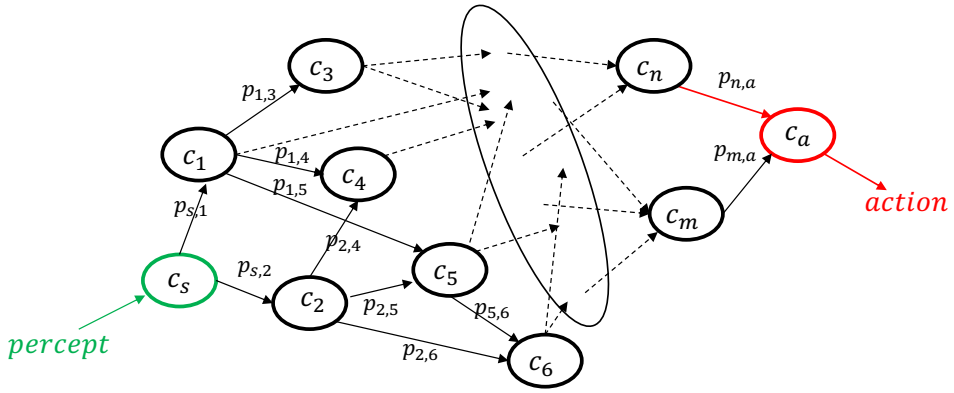


Figure 1: A memory network in PS model and a random walk on the clip space which starts with activation of clip c_s and reaches the action clip c_a that coupled-out the real action. The clips and transition probabilities between them can evolve based on the environment feedback.

The main part of the agent is usually considered to be its learning program which depends on the nature of the agent and its environment. The learning in PS is realized by updating of weights and structure through adding new clips. The connection weights between the clips are updated through Bayesian rules. New clips will be created and added via interaction to the environment as perceptions or from existing clips under certain compositional principles (Melnikov et al., 2017).

2.4 Comparison of PS with Reinforcement Learning

A very brief comparison between PS and other well studied RL algorithms is provided here (see Sutton & Barto, 2018, for details of reinforcement learning schemes), and (see Bjerland, 2015; Mautner et al., 2015, for detailed comparisons of RL and PS). We also discuss our reasons for selecting PS over other RL methods.

It is worth mentioning that RL is different from supervised learning, which is dominant in the field of machine learning. Supervised learning is performed through a set of labeled samples provided by an external supervisor. The objective of this important kind of learning is to generalize or extrapolate the kind of responses that are taught during training. In this way, it can handle situations that were not present in training trials. Reinforcement learning model bears close similarity to the human and animal learning.

The development of reinforcement learning algorithms have benefited from advancements within other fields, specially, psychology and neuroscience. Supervised learning, however, is not an adequate choice for learning from interaction with environment.

The notion of projective simulation can be used as a RL algorithm, since like RL is an independent embodied agent, that interact with environment and learn by trial and error through the feedback. However, PS is a more general framework that is able to use quantum mechanics and solve larger tasks than those possible with RL (Paparo et al., 2014; Mautner et al., 2015).

The most important difference between PS and other standard RL algorithms, is its episodic memory that allows for modeling more complex features. More specifically, learning in RL is based on estimation of value functions, whilst in PS learning is through the re-configuration of memory network. This re-configuration could be simply the update of transition probabilities or by adding/creating new clips. Standard RL has no counterpart for this dynamic change in structure (new clips) which makes PS model more flexible (Melnikov et al., 2017; Bjerland, 2015).

The fact that PS scheme distinguishes between real percepts and actions by using their internal representation makes it more similar to real functioning of the brain, such as the idea of cognitive maps (Tolman, 1948; Behrens et al., 2018), the role of internal manipulation of representations (Piaget et al., 1971), and the brain mechanisms for episodic memory (Hasselmo, 2011).

3 Formation of Stimulus Equivalence Classes in Projective Simulation Setting

In this section, first the standard model of PS formalism is presented in section 3.1 where the notations are mostly from (Melnikov et al., 2017). Then in section 3.2 EPS is presented through algorithms.

3.1 The Formalism of PS

First, the *agent's policy* is defined which is an *external view* of agent's way of behaving at a given time t. The policy is denoted by $P^{(t)}(a|s)$ which represents the probabilities for selecting each possible action $a \in A$, when percept $s \in S$ is received.

Let $C = \{c_1, \ldots, c_p\}$ be the set of possible internal states of the agent. In the clip network of memory, the transition probabilities from clip $c_i \in C$ to clip $c_j \in C$ at time step t is defined as

$$p^{(t)}(c_j|c_i) = \frac{h^{(t)}(c_i, c_j)}{\sum_k h^{(t)}(c_i, c_k)}, \tag{1}$$

where the weight $h^{(t)}(c_i, c_j)$, called h-value is updated as follows at time step t:

$$h^{(t+1)}(c_i, c_j) = h^{(t)}(c_i, c_j) - \gamma\,(h^{(t)}(c_i, c_j) - 1) + \begin{cases} \lambda^{(t)} & \text{if traversed,} \\ 0 & \text{else} \end{cases} \tag{2}$$

where $0 \leq \gamma \leq 1$ is a damping parameter and $\lambda \in \Lambda$ is a non-negative reward given

by the environment. Eq.(2) shows that $h^{(t+1)}(c_i; c_j)$ will be affected by the reward at previous time t, only if the $(c_i; c_j)$ connection was traversed during the random walk at time t. Λ could be a subset of real numbers, in accordance with the learning task and environment type. In the simplest case, $\Lambda = f0; 1g$ where $= 1$ means a reward and $= 0$ means no reward. h-values are initialized with $h_0 = 1$, as soon as a transition link (an edge) is established. A positive damping parameter enables the agent to weaken and even totally forget what it has been learned until time step t (i.e. $h^{(t)}(c_i; c_j)$ h_0). As discussed in PS literature (see Melnikov et al., 2017, for instance), the damping term is not necessary for stationary environments as in contextual bandit tasks (Wang et al., 2005). The SE task that we model has a stationary environment in which the desired percept-action relations do not change over time; however, since we aim at modeling the brain, and as gradual forgetting is an important characteristic of human memory, we keep it in the model.[5]

In order to keep conditional probabilities in Eq.(1) well defined, Eq.(2) guarantees that h-values are lower bounded by h_0 when reward is not negative. An alternative expression for the transition probability, known as the *softmax* (or Boltzmann) distribution function can handle the negative rewards and keeps the transition probabilities non-negative:

$$p^{(t)}(c_j|c_i) = \frac{e^{\ h^{(t)}(c_i;c_j)}}{\sum_k e^{\ h^{(t)}(c_i;c_k)}}; \tag{3}$$

where can be used for tuning the learning rate as well. Lower values of increase the chance of choosing an edge with a larger h-value[6].

Before moving to the next section, we briefly introduce *emotion tags* and *reflection time* in PS model. Emotion tags belong to an emotion space that has arbitrary emotion states. The emotion tags are attached to the transition links between clips and indicate the associated feedback that was stored in the evaluation system of memory. The role of these tags is similar to a short-term memory of the previous rewards for previous actions. So, the agent might avoid an action if it is attached with a negative tag. The state of emotion tag attached to transition links changes based on the feedback; so if the environment changes the agent could update its short-term memory fast. It is important to consider emotion tags, as internal memory of rewards, distinct from external real rewards by environment[7].

The emotion tags can be used by agent in order to avoid immediate action, when the reflection time is bigger than one; $R > 1$. Reflection time is the frequency that agent

[5]To apply PS agent in modeling the contextually controlled equivalence classes, the environment might be considered non-stationary since the established relations could change to new relations. In spite of that, a contextually controlled equivalence class experiment can be considered stationary if one argues that the relations will not change under a specific context.

[6]Note that there is no tuning parameter in Eq.(2) for h-values. Moreover using computationally means that instead of natural logarithm, a different base i.e. e is used.

[7]We do not use emotion tags in the current article. For modeling more advanced scenarios of equivalence formation such as *contextually controlled stimulus equivalence formation* (Bush et al., 1989), one can apply emotion tags to improve the model.

can *reflect* upon its action. More specifically, if the random walk on the memory space ended to an action where agent remembers that the previous reward for this action was not desirable, instead of coupling out the action clip to the real clip, the agent re-excites the percept clip and gives other action clips the chance to be selected.

3.2 Equivalence Projective Simulation Model

Some desired features of a beneficial model in equivalence formation through MTS could be:

1. Ability to form equivalence classes; i.e. be able to correctly match derived relations i.e. symmetry, transitivity and equivalence in MTS trials
2. Ability to show different relatedness factor between stimuli in an equivalence class. For instance, able to show that relatedness is an inverse function of the nodal distance (Fields et al., 1993)
3. Endowment with forgetting ability similar to humans
4. Be able to model memory/learning disabilities by manipulating tuning parameters
5. Possible use as a hypothesis testing tool before making a real experiment

There are different views on mechanism of deriving relations; i.e. either during the training phase and before testing phase or during the MTS test. For instance, in (Galizio et al., 2001) it has been discussed that some degree of equivalence class formation occurs during the MTS training, and that it is further enhanced during the testing. On the other hand, in many studies the emergence of equivalence relations is considered to be only the result of testing lower-stage relations (see, e.g., Dickins, 2015, part E, for a discussion). As explained in (Dickins, 2015), those evidences are established through brain-imaging studies.

At this juncture, we provide the assumptions in our EPS model which can be summarized as follows:

> Appropriate training of baseline relations is necessary for formation of equivalence class, but it is not sufficient.
>
> Any symmetry relation is a function of its entailed baseline relation. K_2 attempts to model mechanisms in the brain that can influence the formation of symmetry.
>
> Formation of transitivity is a function of well trained baseline relations. However, a memory sharpness ( ) less than one, can weaken the effect of baseline relations. Memory sharpness ( ) plays a similar role to K_2 and it rather controls derived relations with nodal distance greater than one; i.e. transitivity and equivalence relations.
>
> could be chosen constant independently of the nodal distance or could vary according to it.

Equivalence formation is a function of both symmetry and transitivity formation. So, K_2 and could be seen as other mechanisms in the brain, along with the reinforcement of baseline relations, that might affect emergence of equivalence relations.

In EPS, whenever symmetry and transitivity relations are emerged, equivalence relations will emerge as well.

EPS does not model reflexivity, since in many experiments with human adults the ability to perform reflexivity task is usually taken for granted (see Dickins et al., 2001, for instance)[8].

The proposed model (EPS) aims at modeling humans behavior where all the stimuli in an equivalence class are expected to be equal. One option is to consider an undirected graph as the memory clip; define all the stimulus-stimulus connections as bidirectional and drop K_2 parameter. However, evidences from experimental studies show that derived relations are sometimes weaker than baseline relations and even not formed at all in some cases. In order to cover more general cases, such as humans that are not able to derive new relations, we consider a directed graph and differentiate between the types of relations. Moreover, EPS can be further extended to model other derived relations in line with equivalence or sameness, as it is in RFT. In such a case, a directed graph, similar to the current selection in EPS, is needed to differentiate relation types.

In the proposed EPS model, the symmetry relations are formed during training, with the assumption that transitivity and equivalence are also formed during training. However, since the response latencies in transitivity and equivalence tests at the beginning are typically longer than trained relations or symmetry tests (Bentall et al., 1993), transitivity and equivalence transition probabilities are calculated for each trial in MTS test. However, by virtue of the flexibility of PS, the model can be modified in a way that the formation of symmetry relations are postponed into the testing phase. On the other hand, one can establish connections in the training phase and gradually update them during MTS testing phase, or during the MTS test.

In the following, we model an arbitrary MTS experiment independent from the training structures (LS, OTM, MTO). The agent has no memory at the beginning i.e. the memory space C is empty, however, all the stimuli potentially belongs to the set of percepts (S) and actions (A), as well as remembered clips C. This initialization will be shown with $S = C = A = ;$. The percept and possible actions are provided by the environment at each time step.

The sample stimuli will make the percept clips and the comparison stimuli will make the action clips. A policy corresponds to a set of stimulus-response rules or associations where S is the set of stimuli and A is the set of responses. The memory space will be updated and enlarged through the training phase. Clips are added the first time that agent perceives them.

[8]PS has the capability to add features to the stimuli and define transitions between clips within each category, including self-loops at each clip. The formation of reflexivity can be achieved using high h-values for self-loops and low h-values within different stimuli in the same category.

The algorithm has two phases, the training phase where the memory network will be shaped and the testing phase where, no new memory clip is created, but new connections can be added and initialized[9].

Training phase

At each time step in general, and at the beginning more specifically, the agent might create new clips, add new transition links and update them based on the reward value. In the model, a memorized clip could simultaneously play the role of either percept clip or action clip.

Since the training structure is through MTS, the possible actions in each trial are limited to a subset of all actions, i.e. the set of comparison stimuli.[10] The action space at time t is denoted by A_t. The probability that action $a^{(t)}$ is chosen by agent when percept $s^{(t)}$ is presented, may depend on the history of experiment. Indeed, the agent learns through changing its internal network, which determines the agent future policy.

In PS model and in general form, the clips as the building blocks of memory are defined as sequences of *remembered* percepts and actions. In modeling the SE, each memory clip represents a remembered stimulus; either as sample stimulus or as comparison stimulus.

Note that the sample stimulus (percept, $s \in S$) and the comparison stimuli (actions $a \in A_t$) belong to different categories, like Greek letters, nature pictures, colored balls, etc. As a result, each of the class members belongs to a different category (say category A, or B, etc.) and there is no connection (paired relation) within elements of a category.

Moreover, in stimulus equivalence, there is no redundancy in the training phase, so the only information that could assist during the learning comes from the members of a category. Consequently, when a new category appears in trials, the agent creates connections with equal weights; since there is no prior information from previous connections. The agent's operation cycle could be summarized as follows:

1. Stimulus $s \in S$ with probability $P^{(t)}(s)$ is perceived from environment.

2. A fixed *input-coupler* probability function $I(c|s)$ activates the memory clip $c \in C$, denoted by c_s . This typically maps the real stimulus s to its internal representation clip with probability 1. For the first time when a stimulus is perceived, a clip will be created and added to the network.

3. Action set A_t is perceived from environment. If any of actions $a \in A_t$ has not an internal image, a clip c_a will be created.

4. If there exist connections among the sample and comparisons, the agent computes the $p^{(t)}(c_a|c_s); a \in A_t$ based on the h-values. If such connections do not exist,

[9]The agent can be provided with the possibility to create new, or "fictitious" clips during the testing phase. Please note that we do not resort to fictitious clips in the current paper.

[10]Please note that, for the sake of simplicity, we consider that the location (order) of comparison stimuli is not important.

agent establishes and initializes them first, and then computes the probabilities $p^{(t)}(c_a|c_s)$.

5. Agent selects one of the possible actions based on the computed probability distribution, then excitation of the selected action clip maps to a real action $a \in A$ through a fixed *output-coupler* function $O(a|c_a)$. Similar to the input coupler function, in general, this function maps the internal action to the real action with probability 1.

6. Then agent receives a positive or negative reward from environment. Connection weights, h-values, will be updated due to this feedback such that the desired match be reinforced.

An important issue in modeling the SE, is that the percepts and actions could play the same role. For instance B_1 is a possible action in AB relation training, and in original PS memory it is remembered as an action clip, but it would play the role of percept in BC training. Subsequently, the role of clips will be changed based on the trial. This double-role of clips, makes the network slightly different from PS. Another distinction between the models is derived relations. Handling symmetry relation in the model is taken care of by establishing the opposite transition links whenever a specific MTS is presented for the first time.

Therefore, initialization of the transition links and h-values for newly added clips is simply done by establishing two direct connections for each possible new match and initializing them with h_0. So if the newly added clip is a percept clip, the number of new connections would be $2|A_t|$. If it is an action clip, just two connections will be established.

To complete the process, the updating rules for h-values based on the environment feedback must be added. Recall that we consider negative reward in the model as well; $\Lambda = \{-1, 0, 1\}$. The reason is that in MTS methods the participants are usually notified whether the chosen stimulus was correct or incorrect.

In the following, two methods for updating h-values will be suggested. The first method, similar to PS, keeps the h-values positive that are lower bounded by h_0. Therefore, this method is suitable for being used in both Eq.(1) and Eq.(3), however the second method can not be used in Eq.(1). The difference between methods occurs when the agent received a negative reward. We will explain the positive reward first and then the two alternatives for negative reward.

Suppose the percept be $s \in S$, coupled into $c_s \in C$ and the chosen action by agent be $a \in A_t$ which is coupled out from clip $c_a \in C$

Let $\lambda^{(t)} = 1$, i.e. the agent chooses the correct option which must be reinforced. The h-value updates will be calculated like PS model i.e.

$$h^{(t+1)}(c_s, c_a) = h^{(t)}(c_s, c_a) \quad (h^{(t)}(c_s, c_a) - h_0) + K_1 \lambda^{(t)}; \tag{4}$$

where K_1 is a positive value, equals unitary based on PS. Moreover, the opposite link, (c_a, c_s) will be updated in a similar way, but with parameter $0 < K_2 < K_1$, see Eq.(5). As mentioned earlier, we could consider a simpler model with

bidirectional connections representing typical humans who are trained well. This is analogous to setting $K_2 = K_1$.

$$h^{(t+1)}(c_a; c_s) = h^{(t)}(c_a; c_s) \qquad (h^{(t)}(c_a; c_s) \quad 1) + K_2 \ ^{(t)}: \tag{5}$$

If $^{(t)} = \quad 1$, i.e. the agent chooses a wrong option which must be inhibited.

- **First scenario for updating h-values**
  This negative reward reinforces all the actions in A_t, except the one that agent has chosen. Let c_{a^θ} 2 O $^1(A_t)$ $fc_a g$; where $O()$ is the output coupler function that transforms a set of clips into real actions. Since $O()$ is one-to-one, its inverse is well defined[11]. The updates rule is:

  $$h^{(t+1)}(c_s; c_{a^\theta}) = h^{(t)}(c_s; c_{a^\theta}) \qquad (h^{(t)}(c_s; c_{a^\theta}) \quad 1) \quad K_3 \ ^{(t)}; \tag{6}$$

  where $K_3 \quad \frac{K_1}{m \quad 1}$; $m = jA_t j$ is the number of options in the action space at time t[12]. Note that, since $^{(t)} = \quad 1$, the term $\quad K_3 \ ^{(t)}$ is positive.
  The symmetry connections are updated in the same way, i.e. the transition weight from clip c_{a^θ} to clip c_s will be increased by an additive factor K_4 where $0 < K_4 \quad \frac{K_2}{m \quad 1}$.

  $$h^{(t+1)}(c_{a^\theta}; c_s) = h^{(t)}(c_{a^\theta}; c_s) \qquad (h^{(t)}(c_{a^\theta}; c_s) \quad 1) \quad K_4 \ ^{(t)}: \tag{7}$$

- **Second scenario for updating h-values**
  The second scenario is similar to the positive reward. The h-values of the transitions will be updated by a negative factor. In this case, only the softmax method can be used for conditional probabilities.

When all clips are created and all possible relations are added and initiated, further training trials are updating the h-values as explained above until the desired relations meet the criterion, so we will be able to move to the testing phase.

Testing phase

The testing phase will be started whenever all training relations meet the mastery criterion. In this phase we test the emergent relations that are not trained explicitly. During the test, basically there is no feedback and we can consider that the evolution of the network based on external feedback is finished. However, one can consider the feedback

[11] We abuse notation since $O()$ coupled-out an action clip to its real counterpart, however, for the sake of simplicity we use the same notation, for the function that sends a set of clips to the set of real actions.

[12] The reason that we define K_3 this way is intuitive. The information we got from the negative reward reinforces other options the same, moreover it is an indirect process so the expectation is to be less effective than the direct ones.

$= 0$ and let the forgetting factor work with dissipation rate . Various testing procedures can be considered by the experimenter such as a random selection of mixture of the learn relations and the emergent ones; or testing symmetry relations first, then transitivity relations, and testing equivalence relations (a combination of symmetry and transitivity) afterwards. At the end of experiment, usually the percentage of correct choices in a specific relation will be calculated and analyzed.

In the artificial model, however, one can use the final policy $P(a|s);\ a \in A;\ s \in S$ for analysis instead of running a testing phase. The agent's functioning during the testing phase can be summarized as follows:

1. Stimulus $s \in S$ with probability $P^{(t)}(s)$ is perceived.

2. A fixed *input-coupler* probability function $I(c_s|s)$ activates the memory clip $c_s \in C$.

3. Action set A_t is perceived from environment.

4. If connections exist among the sample and comparisons, the agent computes the $p^{(t)}(c_a|c_s);\ a \in A_t$ based on the h-values. If such connections do not exist, agent establishes imaginary connections and computes the probabilities $p^{(t)}(c_a|c_s)$. The connections in this case represent the transitivity or equivalence relations[13]. This is the case when *nodal distance* (Fields & Verhave, 1987) or equivalently *nodal number* (Sidman, 1994)[14] is positive, and there is at least a path with length $L \quad 2$ between the possible matches.

   There might be several options and policies to compute the probability of derived connections. For instance one might consider the most probable paths between c_s and each action $c_a; a \in A_t$ which is

$$p^{(t)}(c_a|c_s) = \max_{P_L \in P(c_s;c_a)} \prod_{i=0}^{L-1} p^{(t)}(c_{l_{i+1}}|c_{l_i}); \tag{8}$$

   where $P(c_s;c_a)$ is the set of all possible paths from c_s to c_a, and $P_L \in P(c_s;c_a)$ is a specific one with $L \quad 2$. $l_i;\ i = 1;2; \quad (L \quad 1)$ shows the indices of intermediate clips, while $c_{l_0} = c_s$ and $c_{l_L} = c_a$. In section 4.1 the max-product scenario for computing derived probabilities is addressed.

   Memory sharpness, $0 \qquad 1$, functions as a mechanism to control the formation of transitivity relations, in line with the baseline relations training. Memory sharpness is analogous to the deliberation time in PS model.

   If $= 1$, meaning it is simply removed from the model, the well trained baseline relations result in strong transitivity connections. As we can see, this fact is not

[13] In this case, if one does not establish and update the inverse links during training phase, symmetry connections must be calculated.

[14] A node in equivalence class terms refers to any stimulus, or class member, that connects at least two other members in the equivalence class through training. The nodal distance or nodal number is the number of nodes between the two members.

always true for all real experiments. Therefore we introduce memory sharpness in the model to control transitivity, equivalence relations, and the effect of the nodal distance. Memory sharpness can also represent the effect of comparison stimuli and to what extent agent recalls its memory. Memory sharpness is addressed in section 4.2.

Instead of max-product policy, Eq.(8), one might consider a random walk in C, starting from c_s and ending with a clip in A_t. In other words, instead of finding the most probable path from c_s to each of possibilities in A_t, the probability of reaching each action from c_s can be considered. These probabilities as explained in detail in section 4.3, can be computed easily when actions $c_a \ 2 \ A_t$ are set to be absorbing states of the underlying Markov chain, at time t.

5. Agent selects one of the possible actions based on probabilities $p^{(t)}(c_a|c_s)$ and activation of the action clip maps to a real action $a \ 2 \ A$ through a fixed *output-coupler* function $O(a|c)$.

Since, the aim is to compare performance of this artificial agent with human results, we could just have considered these probabilities without running the testing phase. However, we rather keep it this way to show similar functioning of the agent in the testing phase.

Please look at Algorithm 1 and Algorithm 2 to respectively find a summary of the environment and the agent operations in the training phase. Note that the *Protocol* gives all the information that experimenter (and environment in the artificial model) needs to perform the experiment, including all the stimuli, the training structure (say LS, OTM, or MTO), learning and mastery criterion, etc.

Algorithm 1: Environment operation in EPS model; training phase

input : Experiment Protocol

initialization

$S = ;; \ A = ;; \ t = 1$

begin

  while *All training relations meet the criterion* **do**

    Show the sample stimulus s to the agent

    if $s \not\in S$ **then**
      $S = S \ [\ fsg$

    Show the comparison stimuli $a \ 2 \ A_t$ to the agent

    if $A_t \ 6 \ A$ **then**
      $A = A \ [\ A_t$

    Feedback (reward) to the agent based on its action

    $t = t + 1$

  Show the termination message to the agent

output: $S; \ A$

It is worth mentioning that the training loop in Algorithm 1 might have other stopping criteria along with the mastery of training relations. For instance, an upper bound for number of trials t might be specified in the protocol; or a limitation on the time period that participant can spend before choosing an option. The experimenter might exclude such participants from analysis. However, in the artificial model, there is no need to consider such cases, but instead it is more beneficial to put some restrictions on the memory evolution and tuning parameters to avoid undesired scenarios.

Algorithm 2: Agent operation in EPS model; training phase.

input : Parameters and updating rule

initialization

$C = ;;\ t = 1$

begin

while *Not receiving the termination message* **do**

if $I(s) \in C$ **then**

create $c = I(s)$

$C = C \cup \{c\}$

if $A_t^c = \{a | a \in A_t \text{ and } O^{-1}(a) \in C\} \neq \emptyset$ **then**

for $a \in A_t^c$ **do**

create $c = O^{-1}(a)$

$C = C \cup \{c\}$

Create new connections if any new clip is added; initialize h-values

Compute the probability distribution for $c_a \in A_t$, then choose an action based upon that

Update h-values

$t = t + 1$

output: C

The environment and agent algorithms during the testing phase, that is no feedback,

are presented in Algorithms 3 and 4 respectively.

Algorithm 3: Environment operation in EPS model; testing phase

input : Experiment Protocol, S; A

initialization

$t = 1$

begin

  while *All testing relations presented* **do**

    Show the sample stimulus s to the agent

    Show the comparison stimuli $a \in A_t$ to the agent

    Record the results

    $t = t + 1$

  Show the termination message to the agent

output: Test results

Algorithm 4: Agent operation in EPS model; testing phase.

input : C

initialization

$t = 1$

begin

  while *Not receiving the termination message* **do**

    Receive c_s and $c_a \in A_t$

    if *Connections exist between them* **then**

      Compute probabilities based on h-values

    else

      Compute the probabilities between c_s and $c_a \in A_t$ with an appropriate algorithm

    Choose an option based on the probability distribution over A_t

A sample Protocol is presented in Protocol 1. A description of how EPS models this experiment is provided in detail in Appendix A.

Protocol 1 A sample protocol sheet that experimenter has:

Three, 4-member classes $\{A_1, B_1, C_1, D_1\}$, $\{A_2, B_2, C_2, D_2\}$, and $\{A_3, B_3, C_3, D_3\}$ are going to be trained with arbitrary MTS procedure.

Let the order of training relations be AB, BC, and DC.

The set of comparison stimuli will appear simultaneously after one second delay.

The training is in blocks of 30 trials, a mixture of the possible three relations, each 10 times. Each answer will followed by a feedback, correct ($= 1$) or incorrect ($= \ 1$).[15]

The training mastery criterion is to answer 90% of the trials in the block correctly.

If the participant leaves the experiment, or could not learn a set of relation after $T = 1000$ steps, terminate the experiment by notifying the participant.

If the training mastery criterion is met, the testing phase consists of respectively four blocks; baseline, symmetry, transitivity, and equivalence. Baseline block is composed of $AB; BC;$ and DC relations each 9 times. Symmetry is a block of $BA; CB;$ and CD relations each repeated 9 times. Transitivity block with size 9 contains AC relation. Equivalence block contains $CA; BD; DB; AD$ and DA relations, 9 times each.

Compute the percentage of correct answers for the emergent relations and see if the equivalence relation is formed.

The mastery criterion ratio for the test part is $0{:}9$.

4 Simulation of Stimulus Equivalence

Although, investigation of various parameters' assembly is not in the scope of current paper, in order to validate the model and explain its functionality, some real experiments in the literature, including experiments with patients, are provided and simulated. We have to figure out how parameters must be tuned in order to get similar results to healthy people or patients. We fix $= 1$ in section 4.1 and first simulate the sample experiment provided in Protocol 1 using the max-product method for computing the probability distributions. Next, similar to (Tovar & Westermann, 2017) we simulate some high impact experimental studies in (Sidman & Tailby, 1982), (Devany et al., 1986), and (Spencer & Chase, 1996). The training is in the 'standard' format in which h-values get positive values. A replication of (Spencer & Chase, 1996) with 'softmax' policy, both with positive and negative h-values is reported at the end of this section. In section 4.2, the concept of *memory sharpness* is explained in details. In this section similarities between the *deliberation length* in PS model and nodal distance or nodal number in equivalence theory is discussed, and (Devany et al., 1986) as well as (Spencer & Chase, 1996) studies are modeled. The third case (section 4.3) is to compute the transition probabilities between sample stimulus clip and comparison stimuli clips through a random walk; i.e. as if the action clips are the *absorbing states* of the network. In this setting, similar to (Spencer & Chase, 1996), we explain how the time of reaction might be increased with nodal distance.

[15] For instance, the first block would be a random shuffling of A_1, A_2, and A_3, as sample stimulus, 10 times each. B_1, B_2, and B_3 make the comparison stimuli or action set in a random order.

The concluding experiment is a replication of (Devany et al., 1986) with a different training setting. The aim is to show that EPS is a suitable model to investigate new hypothesis in equivalence study.

The following reported simulation results are the average over 1000 simulations.

4.1 Case 1: Max-Product Algorithm for Probability of Transitive and Equivalence Relations

In the testing phase, when there is no direct connection between percept c_s and actions $c_a \ 2 \ A_t$, in order to find the path from c_s to c_a with the highest probability, one way is to convert the max-product problem into a *min-sum* problem, by using the negative logarithm value.

This is similar to the maximum likelihood algorithms where likelihoods are converted to log likelihoods. In this manner, products are converted to sums and max-products are converted to max-sums. Similarly, the negative log likelihood converts max-product to min-sum. These variations are all trivially equivalent. Through this conversion, finding the path with highest probability will transformed into the lowest-cost path problem. The lowest-cost path problem then can be solved with Dijkstra's algorithm which is an often cited and well known algorithm (Dijkstra, 1959), or min-sum/Viterbi algorithm (see MacKay, 2003, Chapter 16.3, for instance). The final values as probability product of the path with highest probabilities, are normalized to obtain the probability distribution over action set with which an agent uses to select actions.

For more details on how the model computes the max-product of testing phase, see Appendix B.

Note that in this scenario, the relative value of the probabilities is important and the nodal distance that affects the probability values might be ignored during the normalization. In section 4.2 we deal with this issue using memory sharpness parameter.

Experiment 1: Simulation of Protocol 1

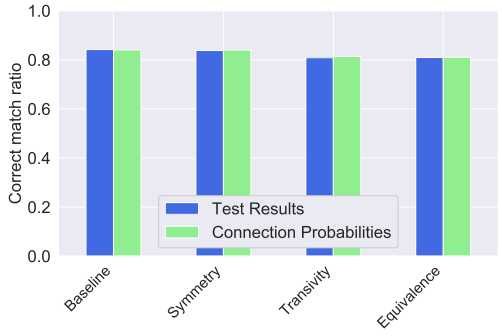
Consider an example based on Protocol 1, where the training phase is AB, BC, and DC respectively. The mastery criterion is set to 0.9 and each block contains 30 trials. For instance, a block for training AB, contains 10 trials with correct match A_1B_1, 10 trials with correct match A_2B_2, and 10 trials with correct match A_3B_3. In the reported results in Figure 2, the blue bars show the outcome of testing phase (the counterpart of what experimenter receive), while the green bars show the connection weights of the memory network at the end of experiment (a representative of the internal state). The Baseline is composed of a block of relations $AB; BC;$ and DC each 9 times, which means each relation is repeated 3 times in the block. Symmetry is a block of $BA; CB;$ and CD each repeated 9 times in a similar way. The Transitivity contains a block of AC relations of size 9. Finally, the Equivalence shows the results for a block of $CA; BD; DB; AD$ and DA relations, 9 times each.

In simulation represented in Figure 2a the parameters are $= 0.001$, $K_1 = 1$, $K_2 = 0.9$, $K_3 = 0.5$, $K_4 = 0.45$. Figure 2a shows that all the relations in equivalence classes are formed. The baseline relations ratio is about $.85$ and for transitivity and equivalence the ratio is about 0.8. In Figure 2b, the forgetting factor changes to

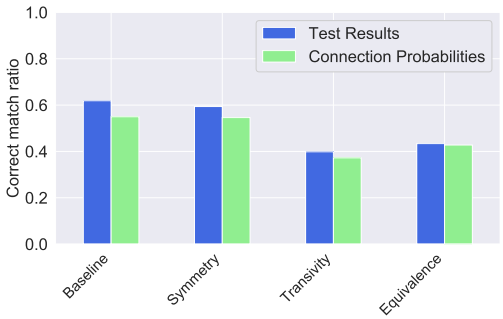
$= 0.01$ which means that agent forgets faster. We see that a higher forgetting factor can affect the results severely. The baseline relations ratio is decreased to 0.6 and transitivity and equivalence relations ratios are decreased to about 0.4. Figure 2b also shows a difference between the connection weights at the end of experiment and the test results. This explains why experimenters usually repeat the relations during training even after mastery, and test the relations as a mixture of all relations to cancel the effect of forgetting.

In Figure 2c, $= 0.001$, $K_1 = 5$, $K_2 = 4$, $K_3 = 2.5$, $K_4 = 2$, we aim at examining how an agent can learn/derive faster by tuning $K_i; i = 1; 2; 3; 4$ parameters. We see that the results shown in Figure 2a and Figure 2c are similar and the only difference is the time for mastery relations, reported in Table 1. With the setting in Figure 2c, each training block have to be repeated around 3.5 in average, whereas this is about 6.5 in Figure 2a setting. So we can tune the block repetition in training by manipulation of parameters. Then, in Figure 2d we study the behavior of an agent when the symmetry relations are not constructed properly by setting $= 0.001$, $K_1 = 20$, $K_2 = 1$, $K_3 = 3$, and $K_4 = 0.3$. We observe that a higher value of K_1 makes training faster, about 1.8 repetition of blocks. As a consequence, the forgetting factor is less effective and the baseline relations ratio is about 0.97. We see that the difference between $K_1 = 20$ and $K_2 = 1$ values resulted into weaker symmetry formation and weaker equivalence relations consequently.

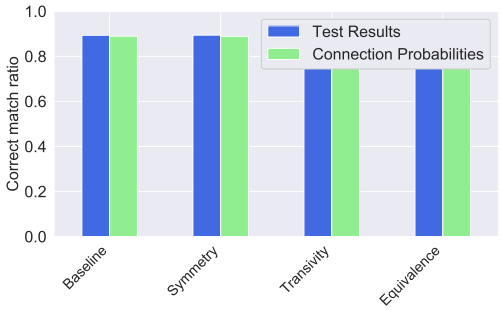
As reported in Table 1 greater value of K_1 makes the training faster in general. The forgetting factor also affects the training time. For instance if $K_1 = 1$ and $= 0.001$, each block must be repeated about 6.5 in average. This will be about 7.3 blocks for $K_1 = 1$ and $= 0.01$, and will be about 1.8 blocks for $K_1 = 20$ and $= 0.001$.



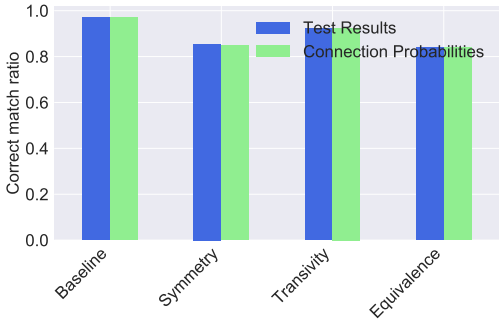
(a) The results for Experiment 1, when = 0:001, $K_1 = 1$, $K_2 = 0{:}9$, $K_3 = 0{:}5$, $K_4 = 0{:}45$



(b) The results for Experiment 1, when = 0:01, $K_1 = 1$, $K_2 = 0{:}9$, $K_3 = 0{:}5$, $K_4 = 0{:}45$



(c) The results for Experiment 1, when = 0:001, $K_1 = 5$, $K_2 = 4$, $K_3 = 2{:}5$, $K_4 = 2$



(d) The results for Experiment 1, when = 0:001, $K_1 = 20$, $K_2 = 1$, $K_3 = 3$, $K_4 = 0{:}3$

Figure 2: Simulation results derived from Experiment 1 with different parameters. The blue bar is the outcome of experiment (analogous to what experimenter receives) and the green bar is the connection weight of the memory network at the end of experiment (representing the internal state)

Table 1: Training time in various settings.

Training	Time (Figure 2a)	Time (Figure 2b)	Time (Figure 2c)	Time (Figure 2d)
(AB, 30)	6.558	7.221	3.470	1.846
(BC, 30)	6.662	7.299	3.476	1.868
(DC, 30)	6.471	7.188	3.350	1.845

In the following, similar to (Tovar & Westermann, 2017) we replicate three studies by (Sidman & Tailby, 1982), (Devany et al., 1986), and (Spencer & Chase, 1996).

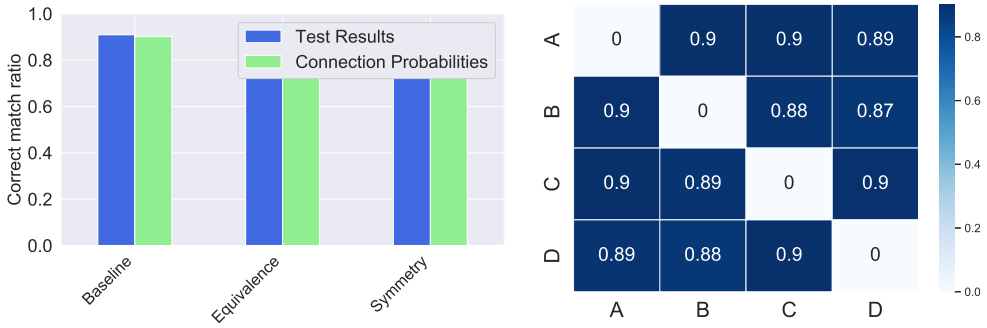
Experiment 2: Sidman and Tailby (1982)

In (Sidman & Tailby, 1982) study, stimulus classes with four members are studied in order to analyze the power of equivalence relations in generation of larger networks.

Eight children with typical development were trained with three 4-member stimulus classes. Stimuli set A were spoken Greek letter names; other stimuli sets (B, C, and D) were sets of different printed Greek letters. The training order was AB and AC relations first and then DC relations. See Table 2 for the order of training and the blocks of MTS trials. The time column shows how many blocks in average used to achieve the mastery in the simulation. We put the mastery criterion ratio to 0.9. The number of necessary blocks are reduced as the relations repeated. The testing phase in (Sidman & Tailby, 1982) experiment is a combination of some baseline and some derived relations, but we test each relation, say AB in a block of 30 trials. The results presented in Figure 3 show the similar results as the experiment, i.e. the formation of relations.

Table 2: The training order in Experiment 2, a replication of (Sidman & Tailby, 1982). The average number of training blocks before reaching the mastery criterion ratio, $0{:}9$, in addition to the results in the last block are reported in Time and Mastery columns respectively.

Training	**Block Size**	**Time**	**Mastery**
1. Training AB			
$A_1B_1; A_2B_2$	20	4.146	0.927
A_1B_1, A_3B_3	20	3.253	0.930
$A_2B_2; A_3B_3$	20	2.077	0.932
$A_1B_1; A_2B_2; A_3B_3$	30	1.641	0.936
2. Training AC			
$A_1C_1; A_2C_2$	20	4.241	0.927
$A_1C_1; A_3C_3$	20	3.244	0.930
$A_2C_2; A_3C_3$	20	2.075	0.934
$A_1C_1; A_2C_2; A_3C_3$	30	1.682	0.935
3. Training AB and AC			
$A_1B_1; A_2B_2; A_3B_3;$ $A_1C_1; A_2C_2; A_3C_3$	30	1.497	0.936
4. Training DC			
$D_1C_1; D_2C_2$	20	4.215	0.929
$D_1C_1; D_3C_3$	20	3.182	0.931
$D_2C_2; D_3C_3$	20	1.991	0.934
$D_1C_1; D_2C_2; D_3C_3$	30	1.628	0.932
3. Training AB, AC and DC			
$A_1B_1; A_2B_2; A_3B_3;$ $A_1C_1; A_2C_2; A_3C_3;$ $D_1C_1; D_2C_2; D_3C_3$	45	1.721	0.935



(a) The agent's results in baseline and derived relations in Experiment 2.

(b) The final probability of correct response between categories.

Figure 3: The replication of (Sidman & Tailby, 1982) when $= 0.001$, $K_1 = 2$, $K_2 = 1.8$, $K_3 = 1$, $K_4 = 0.9$.

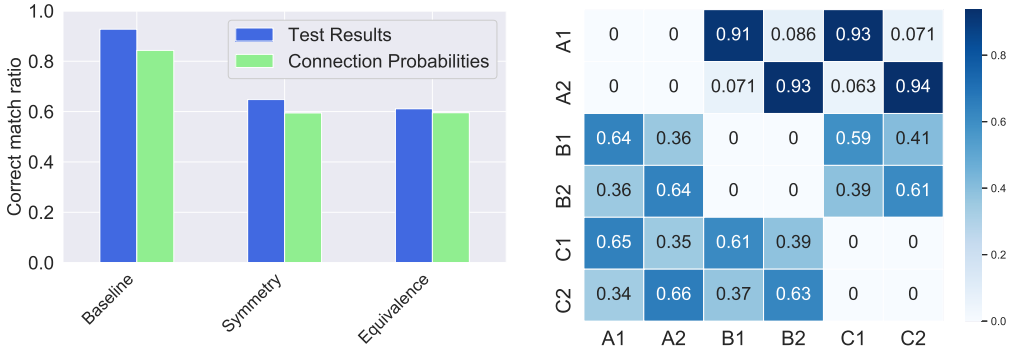
Experiment 3: Devany et. al. (1986)

The results for replication of the experiment in (Devany et al., 1986) is presented here. This is to model the case of language-disabled children who cannot manage the equivalence relations. In (Devany et al., 1986), three groups of children[16], learned AB and AC relations from two classes and they tested for formation of BC and CB. The training order is presented in Table 3. The test results and the transition probabilities of the network at the end of experiment are presented in Figure 4. In the testing phase, each block consists of 20 trials, say BC consists of 10 B_1C_1 and 10 B_2C_2. As results in Figure 4 show, the symmetry and equivalence relations are not formed properly. While the baseline relations' ratio is about 0.9, the BC and CB relations ratio is about 0.6.

Table 3: The training order in Experiment 3, a replication of (Devany et al., 1986) for children with a learning disability without language skills. The Time column shows the average repetition of the training block before reaching the mastery criterion ratio (0.9), when $= 0.01$, $K_1 = 1$, $K_2 = 0.1$, $K_3 = 0.2$, $K_4 = 0.05$. The Mastery column refers to the results in the last block.

Training	**Block Size**	**Time**	**Mastery**
A_1B_1	10	1.825	0.944
A_2B_2	10	1.821	0.947
$A_1B_1; A_2B_2$	10	1.141	0.960
A_1C_1	10	1.871	0.950
A_2C_2	10	1.841	0.949
$A_1C_1; A_2C_2$	10	1.118	0.959
$A_1B_1; A_2B_2; A_1C_1; A_2C_2$	8	1.746	1.000

[16](1) typically developing children, (2) children with a learning disability with some language skills, and (3) children with a learning disability without language skills.



(a) Trained and derived results

(b) Results based on pairs

Figure 4: The results for Experiment 3, the replication of (Devany et al., 1986) when $= 0.01$, $K_1 = 1$, $K_2 = 0.1$, $K_3 = 0.2$, $K_4 = 0.05$.

In this experiment, we weaken the formation of equivalence relations with a lower K_2 parameter which controls the formation of symmetry relation. However, as it can be seen in Experiment 6, even with strongly formed symmetry relations, the formation of of equivalence relations is not guaranteed as non-formation of transitivity induces non-formation of equivalence relations. See Experiment 9 as well to see the effect of K_2 and in the model.

Experiment 4: Spencer and Chase (1996)

The experiment in (Spencer & Chase, 1996) addresses the relatedness (or nodal distance effect) on equivalence formation. It is expected to observe a decrease in the relatedness between the members with higher nodal distance. Spencer and Chase (Spencer & Chase, 1996) measure the response speed during equivalence responding and provide a temporal analysis of the responses. Similar to (Tovar & Westermann, 2017) we try to replicate the *standard group*, formed by college students. However, we measure the relatedness by the ratio of correct answers and the transition probabilities of the network. In the experiment, three 7-member stimulus classes consist of nonsense figures are trained in six sets of relations (AB, BC, CD, DE, EF, and FG) for the three classes) via MTS with three response options per trial. Training consists of seven stages with 48 trials per stage. The training order and the simulation time to learn them are presented in Table 4. The mastery criterion ratio was 0.9. We use three different ordering for the testing phase in which the first two are provided in Table 5 and 6 and the third one is a mixture of all the relations with a random order. Figure 5 shows that the model, similar to real experiments, is sensitive to the order of testing. We have better results for baseline relations, around 0.92, when these relations are tested first in Figures 5a, 5b, 5c, 5d, compared to results in Figure 5e, 5f which is about 0.87. Generally, the forgetting factor affects relations during both the training and testing phases; therefore using a shuffled mix of all relation types in the testing phase can weaken the forgetting effect. For instance, in Figure 5f we see that the strongest relation results are about 0.87 and the weakest relation results are about 0.71, but in Figure 5d these values are respectively 0.92 and 0.6.

Table 4: The training order in Experiment 4, a replication of (Spencer & Chase, 1996) to study the nodal effect. The average time before reaching the mastery criterion ratio (0.9), when $= 0.005$, $K_1 = 5$, $K_2 = 2$, $K_3 = 2$, $K_4 = 1$. The Mastery column refers to the results in the last block.

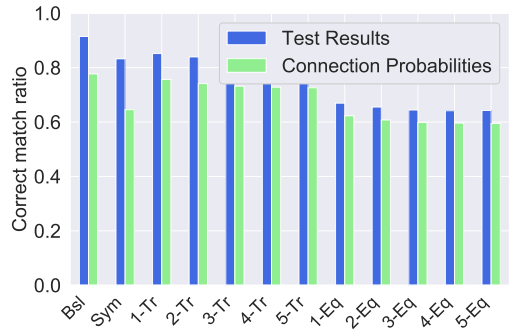
Training	**Number of trials per relation**						**Time**	**Mastery**
	AB	*BC*	*CD*	*DE*	*EF*	*FG*		
AB	48						2.864	0.944
BC	24	24					2.925	0.941
CD	12	12	24				3.139	0.939
DE	9	9	9	24			2.737	0.928
EF	6	6	6	6	24		3.294	0.937
FG	3	3	3	6	9	24	3.438	0.937
Bsl maintenance	3	3	3	3	3	3	1.850	0.964

Table 5: The testing block order in Experiment 4, a replication of (Spencer & Chase, 1996) to study the nodal effect. The results depicted in Figure 5a and Figure 5b.

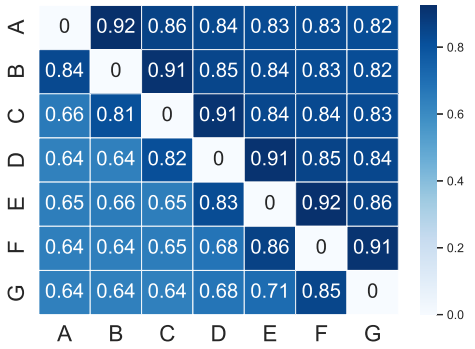
Label	Testing Block	Block size	
Baseline	*AB; BC; CD; DE; EF; FG*	6	9
Symmetry	*BA; CB; DC; ED; FE; GF*	6	9
Transitivity	*AC; AD; AE; AF; AG; BD; BE; BF; BG; CE; CF; CG; DF; DG; EG*	15	9
Equivalence	*CA; DA; EA; FA; GA; DB; EB; FB; GB; EC; FC; GC; FD; GD; GE*	15	9

Table 6: The testing block order in Experiment 4, a replication of (Spencer & Chase, 1996) to study the nodal effect. The results depicted in Figure 5c and Figure 5d.

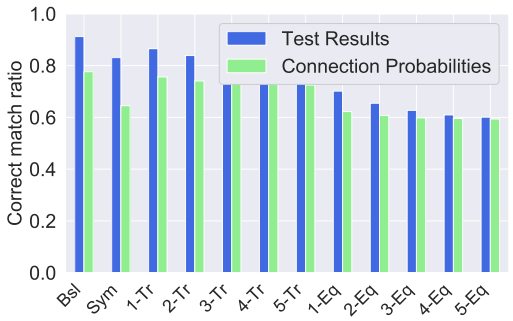
Label		Testing Block	Block size	
Bsl		*AB; BC; CD; DE; EF; FG*	6	9
Sym		*BA; CB; DC; ED; FE; GF*	6	9
1	*Tr*	*AC; BD; CE; DF; EG*	5	9
2	*Tr*	*AD; BE; CF; DG*	4	9
3	*Tr*	*AE; BF; CG*	3	9
4	*Tr*	*AF; BG*	2	9
5	*Tr*	*AG*	1	9
1	*Eq*	*CA; DB; EC; FD; GE*	5	9
2	*Eq*	*DA; EB; FC; GD*	4	9
3	*Eq*	*EA; FB; GC*	3	9
4	*Eq*	*FA; GB*	2	9
5	*Eq*	*GA*	1	9



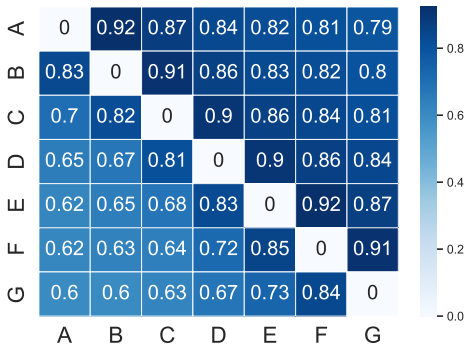
(a) Testing order reported in Table 5.



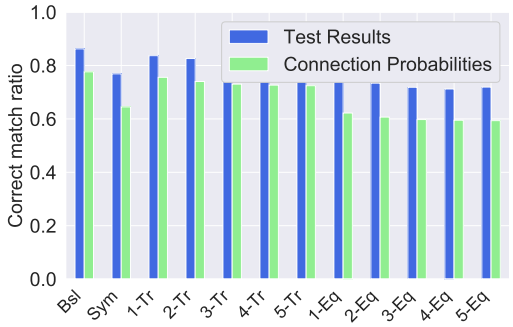
(b) Testing order reported in Table 5.



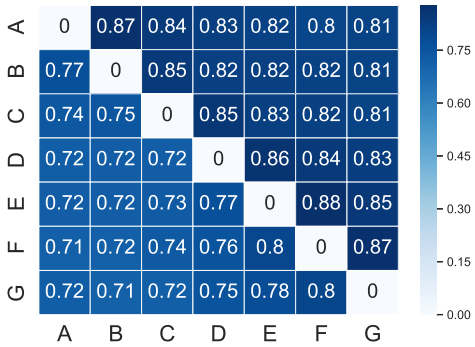
(c) Testing order reported in Table 6.



(d) Testing order reported in Table 6.



(e) Testing order is a mixture of all relations.



(f) Testing order is a mixture of all relations.

Figure 5: Simulation results for Experiment 4, the replication of (Spencer & Chase, 1996) experiment when $= 0.005$, $K_1 = 5$, $K_2 = 2$, $K_3 = 2$, $K_4 = 1$.

Despite the order of testing, the results in Figure 5 show that the model is sensitive to the nodal distance and can show a reverse effect. However, in order to achieve a better nodal effect we simulate this experiment with other methods of computing probability transitions in the testing phase. In the following, we only report the results in the case that the testing is a mixture of all relations.

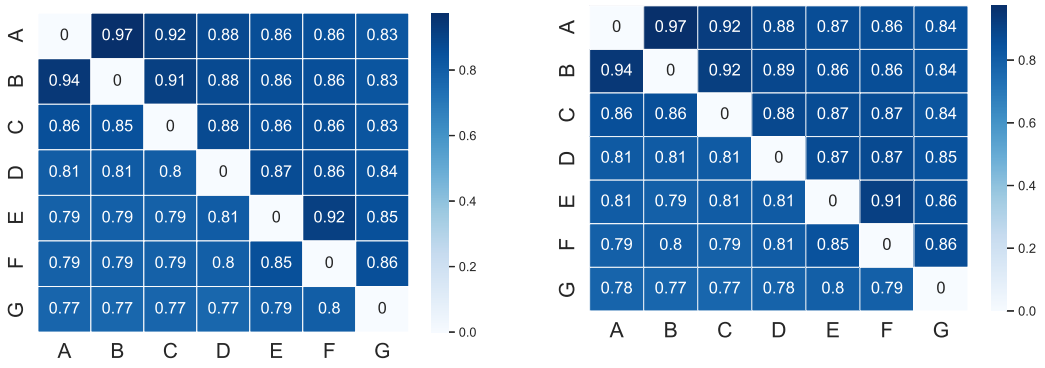
Experiment 5: Using softmax to compute the probabilities

For this experiment, we apply softmax function for transforming h-values into probabilities. In this case, there are two options: first we keep h-values positive and use Eq.(4)-Eq.(7) for updates; second we allow h-values to be negative using Eq.(9) for updates.

$$\begin{array}{ll} h^{(t+1)}(c_i;c_j) = h^{(t)}(c_i;c_j) & (h^{(t)}(c_i;c_j) \quad 1) + K_1 \ ^{(t)}; \quad \text{direct} \\ h^{(t+1)}(c_j;c_i) = h^{(t)}(c_j;c_i) & (h^{(t)}(c_j;c_i) \quad 1) + K_2 \ ^{(t)}; \quad \text{symmetry} \end{array} \tag{9}$$

where $^{(t)} = +1; \quad 1.$

Again, we choose the experiment in (Spencer & Chase, 1996) for replication. In Figure 6a (positive h-values), we see that the higher nodal distance causes weaker results, i.e. the nodal distance and relatedness have a reverse relation. Compare $0{:}97$ for AB with nodal distance zero to $0{:}83$ for AG with nodal distance five. In Figure 6b, we update the h-values using Eq.(9). Figure 6 shows in the case of using softmax, both proposed strategies for updating the h-values work well.



(a) The results for Experiment 5 when = 0:001, = 0:02, $K_1 = 5$, $K_2 = 4$, $K_3 = 2{:}5$, $K_4 = 2$, when h-values are positive.

(b) The results for Experiment 5 when = 0:001, = 0:02, $K_1 = 5$, $K_2 = 4$, when h-values could get negative values.

Figure 6: Simulation results of study (Spencer & Chase, 1996) using softmax function to calculate the transition probabilities.

4.2 Case 2: Different Deliberation Length (Nodal Distance Effect)

One of the scenarios in PS model (Briegel & De las Cuevas, 2012), is to have different deliberation time, where $D = 0$ means direct edges as we have in baseline and symmetry relations, and $D \quad 1$ for sequences with D clips between the percept clip and action clip. Then, after activation of the percept clip, the agent can either directly go to an action clip (called direct) or reach an action clip after some intermediate clips (called compositional). The detailed account of updating connection weights (h-values) can be found in (Briegel & De las Cuevas, 2012). We slightly twist the concept in order to use it in the testing phase of EPS model. The deliberation length could be the counterpart

for nodal distance in equivalence literature.

So, in this scenario during the testing phase and whenever there is no edge between sample stimulus and comparison stimuli, the agent acts as follows:

> Similar to the training phase, if there is no connection between percept and action clips, the agent establishes direct edges and initializes them with h_0.
>
> A *memory sharpness* parameter, 0 1, could control transitivity. It quantifies how much the agent uses the memory i.e. navigates through the memory clips and reaches an action indirectly. The more intact memory the higher value of , and the less intact memory the smaller value of .
>
> In PS model, either an action is chosen through direct connection or compositional clips, the direct connection will be rewarded so the chance to go for direct connections will increase. However, we do not have any reward in this stage and we alternate between $D = 0$ and D 1 using . What we do here is to perform a two-factor selection. First either $D = 0$ or D 1 is chosen based on the Binomial probability with $p =$, then the action will be chosen based on the uniform probabilities ($D = 0$ or no memory) or based on the max-product scenario.
>
> The real probabilities can be simply expressed as a biased sum of the two probabilities:[17]
>
> $$P = P_{D \ 1} + (1 \quad) P_{D=0} \tag{10}$$

As we have mentioned, Case 1 scenario (section 4.1) is a special case of the scenario proposed here. If the memory sharpness factor achieves its maximum value, i.e. $=$ 1, then the direct connections and $D = 0$ has no effect on the chosen option. The reason for differentiation between forgetting factor and the memory sharpness is that in reality, one might not be able to derive new relations, even though direct relations are not forgotten.

Since is expected to somehow control the nodal effect, it could be defined as a function of D. Otherwise, it affects various transitivity and equivalence relations in the same manner, without taking the nodal distance into account. This nodal effect could be fulfilled in several ways. For instance, an effective memory sharpness, say $^{\prime}$, can be defined as a function of both D and . In this way, we have the ordinary forgetting factor ( ), general memory sharpness ( ) that relates to usage of memory and transitivity in general, and effective memory sharpness ($^{\prime}$) which is a sort of memory sharpness under influence of nodal distance.

An effective memory sharpness definition could be $^{\prime} =$ $D(\ ^{\prime})$, where is a fixed value which has already been described. In this case, in order to have $^{\prime}$ 0, we

[17] One might look at this as the effect of memory sharpness on the h-values. Instead of initializing the direct h-values with h_0, they might be initialized with a value K where a smaller is proportional to a bigger value of K (lowering the memory impact and indirect paths). Likewise, a bigger is proportional to a smaller value of K , to scale in favor of using memory and longer paths. Then, the outgoing probabilities will be computed in a similar way as PS.

need ^{0} $\overline{D}$.

Similar method to the Power-Law Model of Psychological Memory can be used as well (Donkin & Nosofsky, 2012); say

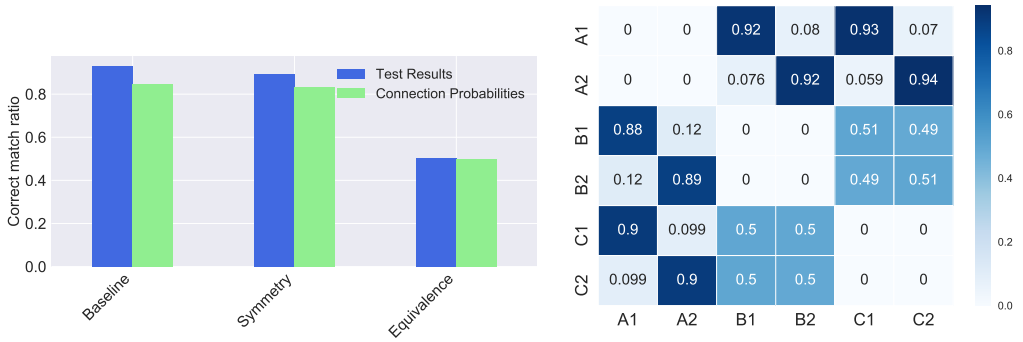
$$^{0} = D \quad ^{\theta}; \text{ for } D \quad 1$$

where 0 ^{0} 1 and the bigger the ^{0} the more intense the nodal effect. In the following and for simplicity, we use the memory sharpness term and symbol for effective memory sharpness as well, unless it is ambiguous.

Experiment 6: Devany et. al. (1986) in Case 2 setting

Here, similar to Experiment 3, the results for replication of the experiment in (Devany et al., 1986) is presented. In Experiment 3, non-formation of symmetry relations causes non-formation of equivalence relations. We show that non-formation of transitivity relations can result into the same case.

The training order is presented in Table 3. The test results and the transition probabilities of the network at the end of experiment are presented in Figure 7. As it can be seen from Figure 7, symmetry relations are derived, but transitivity relations are not formed properly. The baseline relations' ratio is about 0.93, symmetry relations ratio, for BA and CA, is about 0.9, and the BC and CB relations ratio is about 0.5.



(a) Trained and derived results

(b) Results based on pairs

Figure 7: The results for Experiment 6: replication of (Devany et al., 1986) when $=$ 0.01, $K_1 = 1$, $K_2 = 0.9$, $K_3 = 0.5$, $K_4 = 0.45$, $= 0.5$.

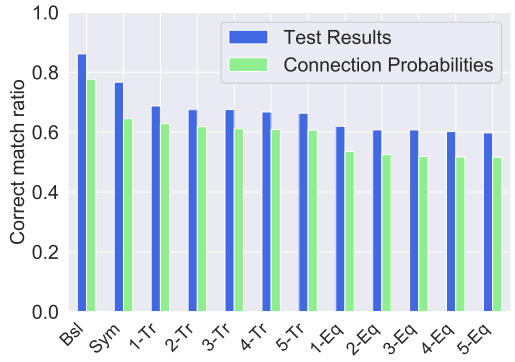
Therefore, in EPS model, formation of equivalence relations is a consequence of formation of both symmetry and transitivity relations.

Experiment 7: Spencer and Chase (1996) in Case 2 setting

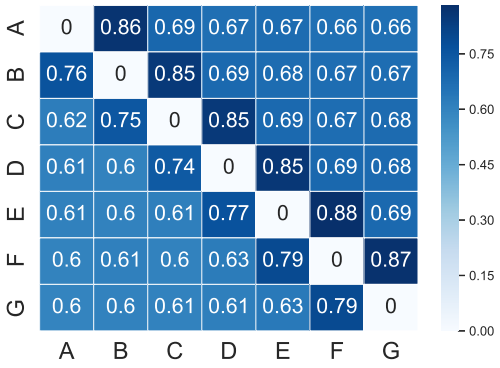
We replicate the experiment in (Spencer & Chase, 1996) to address the relatedness (or nodal distance) using memory sharpness. The training order is presented in Table 4, the testing phase is a mixture of all relations. In Figures 8a and 8b, the memory sharpness is fixed to $= 0.7$. In Figures 8c and 8d, the memory sharpness is adjusted in a linear form ($= 0.7$ $D(0.1)$), and finally in Figures 8e and 8f, the memory sharpness is

adjusted in a power law form ($= 0{:}7 \quad D^{(\ 0{:}8)}$). Comparing the three cases, we observe that in the case of fixed memory sharpness, the indirect relations are influenced in the same way. This can be seen if we compare AC relations with ratio $0{:}69$ to AG relations with ratio $0{:}66$ in Figure 8b. Through adjusting scenarios we can model the nodal effect better, see Figures 8c and 8e and compare them to Figure 8a. In Figure 8d compare the ratio for AC which is $0{:}63$ to the ratio for AG which is $0{:}42$. This rate of changes with nodal distance is much more than fixed memory sharpness. The same comparison in Figure 8f gives $0{:}69$ and $0{:}43$ for nodal distance one at AC and nodal distance five at AG respectively.

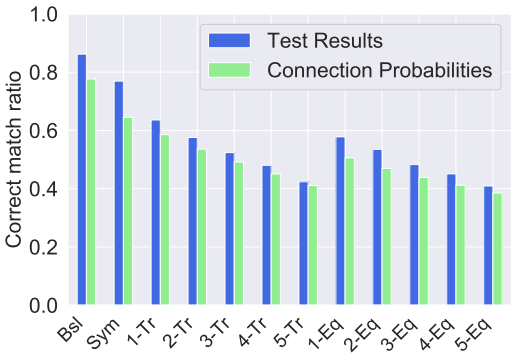
Although the model with memory sharpness (Case 2) is more complex due to extra parameters, it seems that using an adjusting memory sharpness could control the nodal distance and Case 2 sounds more promising.



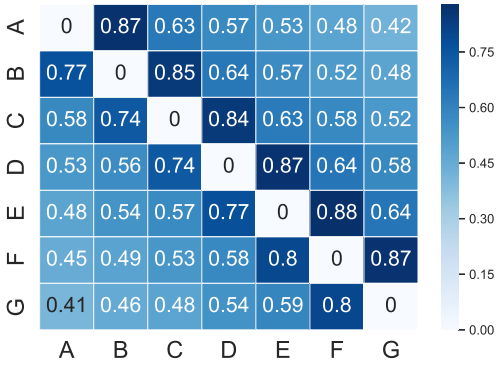
(a) Perceived results by the environment when memory sharpness is fixed: = 0:7.



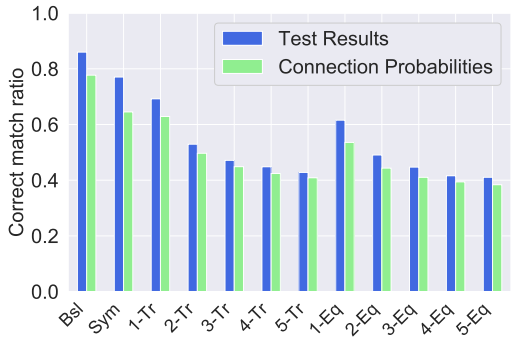
(b) Perceived results by the environment when memory sharpness is fixed: = 0:7.



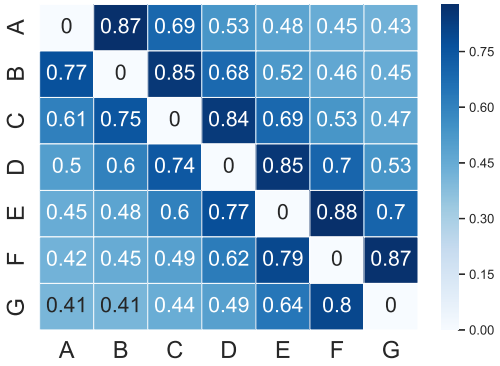
(c) Perceived results by the environment when memory sharpness is linearly adjusting with nodal distance: = 0:7 $D(0{:}1)$.



(d) Perceived results by the environment when memory sharpness is linearly adjusting with nodal distance: = 0:7 $D(0{:}1)$.



(e) Perceived results by the environment when memory sharpness is adjusting with nodal distance using power law: = 0:7 $D^{(\ 0:8)}$.



(f) Perceived results by the environment when memory sharpness is adjusting with nodal distance using power law: = 0:7 $D^{(\ 0:8)}$.

Figure 8: Simulation results for Experiment 7, replication of study (Spencer & Chase, 1996), when $= 0{:}005$, $K_1 = 5$, $K_2 = 2$, $K_3 = 2$, $K_4 = 1$ with different memory sharpness values.

4.3 Case 3: Action Set as the Set of Absorbing States

In the standard PS model, an action is coupled out whenever the relevant action clip is reached. If a unit transition probability is assigned from each action clip to itself, then the action clips will be absorbing states of the Markov chain (or memory clip network). Briefly, in an absorbing Markov chain it is impossible to leave some states once visited. Those states are called absorbing states. Moreover, any state has a path to reach such a state. The non-absorbing states in an absorbing Markov chain are transient. In our equivalence PS, since a clip can be used as percept clip and action clip interchangeably, the network does not have absorbing states in its general form. However, for simplicity, at each trial where the agent perceives the percept and the action set, we ignore the output connections and assign a unit transition probability for the clips in the action clip. As a result, the clips in the action set temporarily become the absorbing states.

This way, instead of using transition probabilities, we consider the probability of being absorbed by an action clip in A_t, starting at percept stimulus. This is more close to the logic of PS memory clip and the random walk[18].

If the size of non-absorbing or transient clips in the network be n_t and the number of absorbing states be $m = jA_tj$, the transition matrix of the network can be written as:

$$P = \begin{matrix} Q & R \\ 0 & I_m \end{matrix}$$

where Q is an n_t n_t matrix, R is an n_t m matrix, 0 is the m n_t zero matrix, and I_m is the identity matrix of size m m. The fundamental matrix is defined as

$$N = (I_{n_t} \quad Q)^{-1} = \sum_{k=0}^{\infty} Q^k.$$

If one starts at clip i, the expected number of nodes before entering an action clip is the ith component of the vector $N1$. This could be used to address the answering time that is mentioned in (Spencer & Chase, 1996). The probability starting at i and ending at absorbing state j is the $(i; j)$th entry of matrix $M = NR$.

Note that as mentioned in the original PS model, it is possible that the random walk on the clip space falls in a loop and for instance goes back and forth between two clips that have high transition probability to each other. As we will see in the simulation, this results into a larger expected steps. However, various mechanisms could control this undesired situation. A method which is stated in (Briegel & De las Cuevas, 2012) is to put a limitation on the random walk time, called *maximum deliberation time* $D_{\max}$. If the agent could not manage to reach an action before $D_{\max}$, whatever the ultimate action would be, it will not be rewarded. Since we are using the random walk for the testing phase, this is not applicable though. Even if we use the absorbing Markov chain

[18] One might bias the random walk according to the action set. Since it would be different if the actions are present simultaneously, or they are given with a delay (in the delay case, the random walk will be start without any bias from actions presence). In other words, presence of the action set plays a reinforcing role. One possibility is to consider a parameter similar to memory sharpness that controls the effect of action set.

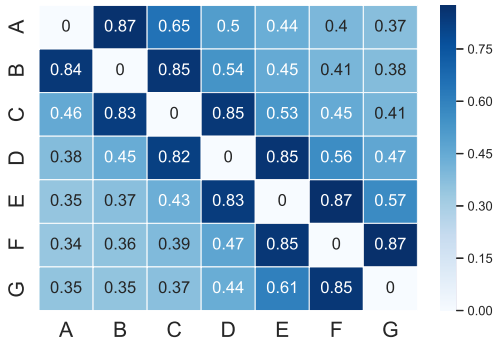
to compute the probabilities during the training, instead of just relying on direct connections, $D_{\max}$ is not a compatible strategy with real experiments. Since in the standard SE protocols, too much delay does not have the penalty of not receiving feedback from experimenter (here environment). One might use the concept of gating in the model which is used for instance in long-short term memories (Hochreiter & Schmidhuber, 1997) or a kind of local emotional tags similar to PS. Another option to avoid revisit clips could be Self-Avoiding Walks (SAWs)[19].

Experiment 8: Spencer and Chase (1996) in Case 3 Setting (Absorbing States)

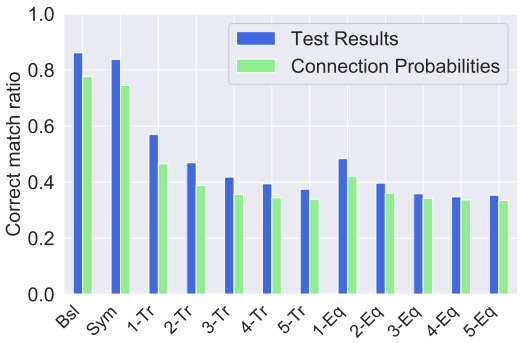
We replicate the experiment in (Spencer & Chase, 1996) to address the relatedness in absorbing state setting. The training order is presented in Table 4. Note that in this experiment, we use a second measurement of nodal distance which is the expected number of transitions between the sample stimulus and an action.

Figure 8a and Figure 8b show that computing probabilities in an absorbing Markov chain setting has the capability to show a sort of nodal effect. Compare AB with 0.87 to AG with 0.37. Figure 8c shows that in general, greater nodal distance causes higher expected steps. However, based on the results, nodal distance is not the only factor on expected steps and the probability distribution plays a stronger role. First, we note that for AB and GF the expected number of steps is one. That's because A and G are located at the two end sides of the learning series. On the other hand, the expected number of steps for BA with zero nodal distance is around 23, which is more than expected number of steps for BE with nodal distance two (around 12). So, we observe that nodal distance is not the only effect; but the input and output probabilities, and the location of a category in the learning order are also important. Compare BA and FE where both are symmetry relations and one node away from one end of the series, but the expected number of steps for FE is around 4, which is much less than 23 steps of BA. So, the general form of the network must be taken into account as there are many studies on differences of LS, OTM, and MTO training structures (see, e.g., Arntzen et al., 2010a; Arntzen & Hansen, 2011; Arntzen, 2012).

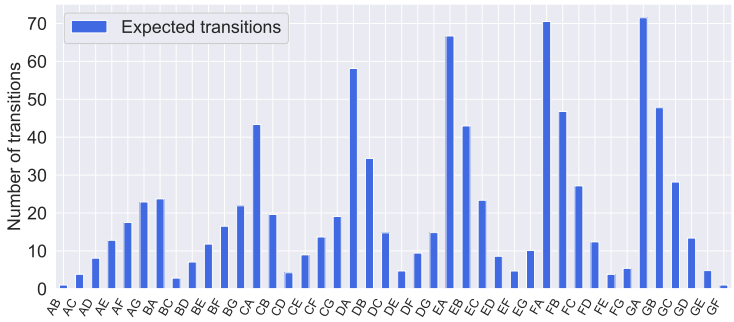
[19] Note that unlike the random walk, the SAW is not a Markovian stochastic process.



(a) The ratio of correct matches perceived by environment.



(b) The results based on nodal distance.



(c) The expected number of steps between the sample stimulus and a match in comparison stimuli.

Figure 9: Simulation results of study (Spencer & Chase, 1996) using absorbing markov chain (Experiment 8) when $= 0{:}005$, $K_1 = 5$, $K_2 = 4$, $K_3 = 2{:}5$, $K_4 = 2$.

In Figure (9c) we see that the expected number of steps is higher than the shortest path that shows back and forth transitions between the clips.

Experiment 9: Devany et. al. (1986) with a New Training Setting

Computational models can be used to gain insights, build hypotheses, make predictions, and formulate questions that lead to new directions for empirical research. Experiment 9, could be considered as a hypothetical experiment to see how the proposed EPS model can interact with practical experiments. The question is whether it is possible to gain better results in (Devany et al., 1986) with the same amount of trials but with a different training order. We propose a new training order that presented in Table 7, along with the original one in (Devany et al., 1986).

Table 7: The proposed training order in Experiment 9, an alternative training order to (Devany et al., 1986).

Original Training	**Suggested Training**	**Block Size**
A_1B_1	A_1B_1	10
A_2B_2	$\mathbf{B_2A_2}$	10
$A_1B_1; A_2B_2$	$A_1B_1; \mathbf{B_2A_2}$	10
A_1C_1	A_1C_1	10
A_2C_2	$\mathbf{C_2A_2}$	10
$A_1C_1; A_2C_2$	$A_1C_1; \mathbf{C_2A_2}$	10
$A_1B_1; A_2B_2; A_1C_1; A_2C_2$	$A_1B_1; \mathbf{B_2A_2}; A_1C_1; \mathbf{C_2A_2}$	8

(a) Results with original training order, when = 1

(b) Results with new training order, when = 1

(c) Results with new training, when = 0:5

Figure 10: The results for Experiment 9, when we use similar parameters as Experiment 3; = 0:01, $K_1 = 1$, $K_2 = 0{:}1$, $K_3 = 0{:}2$, $K_4 = 0{:}05$, but a different memory sharpness in 10c.

Throughout Experiment 9, we suggest that if one chooses a mixture of trials between the given categories, the symmetry relations will be stronger and as a consequence the equivalence relations might be formed. Intuitively, by reinforcing one of the pairs (say A_1B_1), the other one will be inhibited A_1B_2 and so A_2B_2 gets a higher chance to be selected. The idea is to train B_2A_2 which is not formed well (with the chosen parameter values), instead of training A_2B_2 that might be derived easier. The same argument shows that B_2A_2 training, accelerates deriving B_1A_1.

A comparison between Figure 10a and Figure 10b, shows that the agent with similar parameters and training time can achieve a better results in BC and CB relations; compare $0{:}6$ in Figure 10a for these relations to $0{:}9$ in Figure 10b. Based on these results, EPS could suggest experimenters to consider a different combination of trials in the baseline training blocks.

To complete the circle, suppose an experimenter tests out this hypothesis in practice and observes that still the equivalence relations are not formed. This means that the problem in equivalence formation does not emanate solely from symmetry relation formation, but maybe from transitivity formation that we referred to in Experiment 6. In Figure 10c we put $= 0{:}5$ in order to model the new results. This time, even though the symmetry relations are formed well, the equivalence relations are not formed (around $0{:}64$). So, first we study the effect of a new training procedure in this experiment, and

then emphasize the fact that equivalence relation formation in EPS model is based on both symmetry and transitivity relations. In Experiment 3, the lack of strong symmetry relations results in weak equivalence relations (Figure 10a). We suggest a possible solution by redesigning the training setting in Experiment 9. However, if the transitivity relations are not formed similar to Experiment 6, equivalence relations can not be derived as well (Figure 10c).

Experiment 9, is an example on the possibility to generate and vet an idea in equivalence theory prior to full experimental testing. Note that, to study a behavior, the most important thing is to tune parameters of the model, then use them to study new settings.

5 Concluding Remarks

Although computational models of cognition and behavior are simplified versions of the brain activity, they might be a useful tool to study brain activity and to analyze experimental data. In this regard, the model must be interpretable and biologically plausible so that psychologists can rely on.

In the current study, we propose a machine learning scheme for modeling the equivalence formation. To the best of our knowledge, it is the first study that approaches computational modeling in stimulus equivalence through machine learning. We consider a specific reinforcement learning model, projective simulation, as the ground of our model, since we found this model flexible and more adaptable to equivalence class formation. The model has an internal episodic memory that could easily be interpreted and extended to replicate various stimulus equivalence experimental settings. The proposed model in this study is not a black-box model which makes it more appropriate for researchers in behavior analysis to accept and apply it. As discussed in the simulation results, the model can control various factors such as learning rate, forgetting rate, symmetry and transitivity formation. Nodal effect, which is an important topic in equivalence formation, is simulated and explained with EPS. Through Simulation of some real experiments in behavior analysis literature, we display the model capability to behave like typical participants or participants with special disabilities. Moreover, we show that how a research idea in equivalence theory can be studied through EPS. The proposed simulations can be considered as a proof of concept; but studying the parameters, optimal tuning for a specific behavior, comparing the proposed calculation of probabilities, require a separate study. For instance, one might tune different parameters to model a specific behavior in MTS trials or study the optimal number of members and categories, comparing LS, OTM, and MTO, and so on. Using softmax function to calculate probabilities in absorbing Markov chain model as well as adding memory sharpness effect are straightforward steps. Furthermore, it is possible to add direct edges, initialize them (with h_0 or an adjusting h-value that is proportional to other output h-values) and then compute the absorbing probabilities. Alternative options are using emotional tags, gating, or self-avoiding random walks. The main advantage of using PS as the ground model is that it is quite flexible and easy to interpret. It can be modified in order to address other types of training procedures, such as compound stimuli, instead of MTS. A possible approach for modeling compound stimuli is to use the generalized projective simulation (Melnikov et al., 2017) that considers clips composed

of different categories. On the other hand, EPS model can be considered as an extension of PS model that might be interesting solely from machine learning point of view. For instance, symmetry connections and variable action sets could be used in more general applications. Overall, we believe that PS framework in general, and the introduced EPS model in specific, could be a powerful and flexible tool for computational modeling in equivalence theory that has many advantages over the existing connectionist models.

Acknowledgments

We thank Mahdi Rouzbahaneh for his help and comments on the Python code and for generously writing a graphical user friendly interface for our simulator that is easy to use by behavior analysts researchers. The source code of EPS model simulator is available for download at github.

References

Arntzen, E. (2012). Training and testing parameters in formation of stimulus equivalence: Methodological issues. *European Journal of Behavior Analysis*, 13(1), 123–135.

Arntzen, E., Grondahl, T., & Eilifsen, C. (2010a). The effects of different training structures in the establishment of conditional discriminations and subsequent performance on tests for stimulus equivalence. *The Psychological Record*, 60(3), 437–461.

Arntzen, E., Halstadtro, L.-B., Bjerke, E., & Halstadtro, M. (2010b). Training and testing music skills in a boy with autism using a matching-to-sample format. *Behavioral Interventions: Theory & Practice in Residential & Community-Based CIinical Programs*, 25(2), 129–143.

Arntzen, E., Halstadtro, L.-B., Bjerke, E., Wittner, K. J., & Kristiansen, A. (2014). On the sequential and concurrent presentation of trials establishing prerequisites for emergent relations. *The Behavior Analyst Today*, 14(1-2), 1.

Arntzen, E. & Hansen, S. (2011). Training structures and the formation of equivalence classes. *European Journal of Behavior Analysis*, 12(2), 483–503.

Arntzen, E. & Holth, P. (1997). Probability of stimulus equivalence as a function of training design. *The Psychological Record*, 47(2), 309–320.

Arntzen, E., Steingrimsdottir, H., & Brogård-Antonsen, A. (2013). Behavioral studies of dementia: Effects of different types of matching-to-sample procedures. *Eur J Behav Anal*, 40, 17–29.

Arntzen, E. & Steingrimsdottir, H. S. (2014). On the use of variations in a delayed matching-to-sample procedure in a patient with neurocognitive disorder. *Mental Disorder; Swahn, MH, Palmier, JB, Braunstein, SM, Eds*, (pp. 123–138).

Arntzen, E. & Steingrimsdottir, H. S. (2017). Electroencephalography (eeg) in the study of equivalence class formation. an explorative study. *Frontiers in human neuroscience*, 11, 58.

Baddeley, A. D., Kopelman, M. D., & Wilson, B. A. (2003). *The handbook of memory disorders*. John Wiley & Sons.

Barnes, D. (1994). Stimulus equivalence and relational frame theory. *The Psychological Record*, 44(1), 91–125.

Barnes, D. & Hampson, P. J. (1993). Stimulus equivalence and connectionism: Implications for behavior analysis and cognitive science. *The Psychological Record*, 43(4), 617–638.

Barnes, D. & Hampson, P. J. (1997). Connectionist models of arbitrarily applicable relational responding: A possible role for the hippocampal system. In *Advances in Psychology*, volume 121 (pp. 496–521). Elsevier.

Barnes, D. & Holmes, Y. (1991). Radical behaviorism, stimulus equivalence, and human cognition. *The Psychological Record*, 41(1), 19–31.

Barnes-Holmes, D., Barnes-Holmes, Y., & Cullinan, V. (2000). Relational frame theory and skinners verbal behavior: A possible synthesis. *The Behavior Analyst*, 23(1), 69–84.

Barnes-Holmes, S. C. H. D. & Roche, B. (2001). *Relational frame theory: A post-Skinnerian account of human language and cognition*. Springer Science & Business Media.

Bechtel, W. & Abrahamsen, A. (1991). *Connectionism and the mind: An introduction to parallel processing in networks.* Basil Blackwell.

Behrens, T. E., Muller, T. H., Whittington, J. C., Mark, S., Baram, A. B., Stachenfeld, K. L., & Kurth-Nelson, Z. (2018). What is a cognitive map? organizing knowledge for flexible behavior. *Neuron*, 100(2), 490–509.

Bentall, R. P., Dickins, D. W., & Fox, S. R. (1993). Naming and equivalence: Response latencies for emergent relations. *The Quarterly Journal of Experimental Psychology*, 46(2), 187–214.

Bjerland, Ø. F. (2015). Projective simulation compared to reinforcement learning. Master's thesis, The University of Bergen.

Bódi, N., Csibri, É., Myers, C. E., Gluck, M. A., & Kéri, S. (2009). Associative learning, acquired equivalence, and flexible generalization of knowledge in mild alzheimer disease. *Cognitive and Behavioral Neurology*, 22(2), 89–94.

Briegel, H. J. & De las Cuevas, G. (2012). Projective simulation for artificial intelligence. *Scientific reports*, 2, 400.

Brogård-Antonsen, A. & Arntzen, E. (2019). Analyzing conditions for recognizing pictures of family members in a patient with alzheimer's disease. *Behavioral Interventions*.

Bush, K. M., Sidman, M., & Rose, T. d. (1989). Contextual control of emergent equivalence relations. *Journal of the Experimental Analysis of Behavior*, 51(1), 29–45.

Clayton, M. C. & Hayes, L. J. (1999). Conceptual differences in the analysis of stimulus equivalence. *The Psychological Record*, 49(1), 145–161.

Commons, M. L., Grossberg, S., & Staddon, J. (2016). *Neural network models of conditioning and action*. Routledge.

Cullinan, V. A., Barnes, D., Hampson, P. J., & Lyddy, F. (1994). A transfer of explicitly and nonexplicitly trained sequence responses through equivalence relations: An experimental demonstration and connectionist model. *The Psychological Record*, 44(4), 559–585.

Debert, P., Huziwara, E. M., Faggiani, R. B., De Mathis, M. E. S., & McIlvane, W. J. (2009). Emergent conditional relations in a go/no-go procedure: Figure–ground and stimulus-position compound relations. *Journal of the Experimental Analysis of Behavior*, 92(2), 233–243.

Debert, P., Matos, M. A., & McIlvane, W. (2007). Conditional relations with compound abstract stimuli using a go/no-go procedure. *Journal of the Experimental Analysis of Behavior*, 87(1), 89–96.

Devany, J. M., Hayes, S. C., & Nelson, R. O. (1986). Equivalence class formation in language-able and language-disabled children. *Journal of the experimental analysis of behavior*, 46(3), 243–257.

Dickins, D. (2015). *Stimulus Equivalence: A Laboratory Artefact or the Heart of Language?* PhD thesis, University of Huddersfield.

Dickins, D. W., Singh, K. D., Roberts, N., Burns, P., Downes, J. J., Jimmieson, P., & Bentall, R. P. (2001). An fmri study of stimulus equivalence. *Neuroreport*, 12(2), 405–411.

Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1), 269–271.

Donkin, C. & Nosofsky, R. M. (2012). A power-law model of psychological memory strength in short-and long-term recognition. *Psychological Science*, 23(6), 625–634.

Ducatti, M. & Schmidt, A. (2016). Learning conditional relations in elderly people with and without neurocognitive disorders. *Psychology & Neuroscience*, 9(2), 240.

Fields, L., Adams, B. J., & Verhave, T. (1993). The effects of equivalence class structure on test performances. *The Psychological Record*, 43(4), 697–712.

Fields, L., Adams, B. J., Verhave, T., & Newman, S. (1990). The effects of nodality on the formation of equivalence classes. *Journal of the Experimental Analysis of behavior*, 53(3), 345–358.

Fields, L. & Verhave, T. (1987). The structure of equivalence classes. *Journal of the experimental analysis of behavior*, 48(2), 317–332.

Fienup, D. M., Covey, D. P., & Critchfield, T. S. (2010). Teaching brainbehavior relations economically with stimulus equivalence technology. *Journal of Applied Behavior Analysis*, 43(1), 19–33.

Fienup, D. M., Wright, N. A., & Fields, L. (2015). Optimizing equivalence-based instruction: Effects of training protocols on equivalence class formation. *Journal of Applied Behavior Analysis*, 48(3), 613–631.

Fodor, J. A. & Pylyshyn, Z. W. (1988). Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28(1-2), 3–71.

Galizio, M., Stewart, K. L., et al. (2001). Clustering in artificial categories: An equivalence analysis. *Psychonomic Bulletin & Review*, 8(3), 609–614.

Gallagher, S. M. & Keenan, M. (2009). Stimulus equivalence and the mini mental status examination in the elderly. *European Journal of Behavior Analysis*, 10(2), 159–165.

Grisante, P. C., Galesi, F. L., Sabino, N. M., Debert, P., Arntzen, E., & McIlvane, W. J. (2013). Go/no-go procedure with compound stimuli: effects of training structure on the emergence of equivalence classes. *The Psychological Record*, 63(1), 63–72.

Groskreutz, N. C., Karsina, A., Miguel, C. F., & Groskreutz, M. P. (2010). Using complex auditory-visual samples to produce emergent relations in children with autism. *Journal of Applied Behavior Analysis*, 43(1), 131–136.

Hall, G. A. & Chase, P. N. (1991). The relationship between stimulus equivalence and verbal behavior. *The Analysis of Verbal Behavior*, 9(1), 107–119.

Hasselmo, M. E. (2011). *How we remember: brain mechanisms of episodic memory*. MIT press, Cambridge Massachusetts.

Hayes, S. (1994). Relational frame theory: A functional approach to verbal behavior. *Behavior analysis of language and cognition*, (pp. 11–30).

Hayes, S. C. (1989). Nonhumans have not yet shown stimulus equivalence. *Journal of the Experimental Analysis of behavior*, 51(3), 385–392.

Hayes, S. C. (1991). A relational control theory of stimulus equivalence. *Dialogues on verbal behavior*, (pp. 19–40).

Hayes, S. C. & Sanford, B. T. (2014). Cooperation came first: Evolution and human cognition. *Journal of the Experimental Analysis of Behavior*, 101(1), 112–129.

Hayes, S. C., Sanford, B. T., & Chin, F. T. (2017). Carrying the baton: Evolution science and a contextual behavioral analysis of language and cognition. *Journal of contextual behavioral science*, 6(3), 314–328.

Hochreiter, S. & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8), 1735–1780.

Hove, O. (2003). Differential probability of equivalence class formation following a one-to-many versus a many-to-one training structure. *The Psychological Record*, 53(4), 617–634.

Lantaya, C. A., Miguel, C. F., Howland, T. G., LaFrance, D. L., & Page, S. V. (2018). An evaluation of a visual–visual successive matching-to-sample procedure to establish equivalence classes in adults. *Journal of the experimental analysis of behavior*, 109(3), 533–550.

Lew, S. E. & Zanutto, S. B. (2011). A computational theory for the learning of equivalence relations. *Frontiers in human neuroscience*, 5, 113.

Lovett, S., Rehfeldt, R. A., Garcia, Y., & Dunning, J. (2011). Comparison of a stimulus equivalence protocol and traditional lecture for teaching single-subject designs. *Journal of Applied Behavior Analysis*, 44(4), 819–833.

Lyddy, F. & Barnes-Holmes, D. (2007). Stimulus equivalence as a function of training protocol in a connectionist network. *The Journal of Speech and Language Pathology–Applied Behavior Analysis*, 2(1), 14.

Lyddy, F., Barnes-Holmes, D., & Hampson, P. J. (2001). A transfer of sequence function via equivalence in a connectionist network. *The Psychological Record*, 51(3), 409–428.

MacKay, D. J. (2003). *Information theory, inference and learning algorithms*. Cambridge university press.

Markham, M. R. & Dougher, M. J. (1993). Compound stimuli in emergent stimulus relations: Extending the scope of stimulus equivalence. *Journal of the Experimental Analysis of Behavior*, 60(3), 529–542.

Mautner, J., Makmal, A., Manzano, D., Tiersch, M., & Briegel, H. J. (2015). Projective simulation for classical learning agents: a comprehensive investigation. *New Gener. Comput.*, 33(1), 69–114.

McClelland, J. L. (2009). The place of modeling in cognitive science. *Topics in Cognitive Science*, 1(1), 11–38.

McClelland, J. L., Rumelhart, D. E., Group, P. R., et al. (1987). *Parallel distributed processing*, volume 2. MIT press Cambridge, MA:.

McDonagh, E., McIlvane, W., & Stoddard, L. T. (1984). Teaching coin equivalences via matching to sample. *Applied Research in Mental Retardation*, 5(2), 177–197.

McLay, L. K., Sutherland, D., Church, J., & Tyler-Merrick, G. (2013). The formation of equivalence classes in individuals with autism spectrum disorder: A review of the literature. *Research in Autism Spectrum Disorders*, 7(2), 418–431.

Melnikov, A. A., Makmal, A., Dunjko, V., & Briegel, H. J. (2017). Projective simulation with generalization. *Scientific reports*, 7(1), 14430.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. (2015). Human-level control through deep reinforcement learning. *Nature*, 518(7540), 529.

Murre, J. M., Graham, K. S., & Hodges, J. R. (2001). Semantic dementia: relevance to connectionist models of long-term memory. *Brain*, 124(4), 647–675.

Ninness, C., Ninness, S. K., Rumph, M., & Lawson, D. (2018). The emergence of stimulus relations: human and computer learning. *Perspectives on Behavior Science*, 41(1), 121–154.

Ninness, C., Rehfeldt, R. A., & Ninness, S. K. (2019). Identifying accurate and inaccurate stimulus relations: Human and computer learning. *The Psychological Record*, (pp. 1–24).

Nissen, H. W. (1951). Analysis of a complex conditional reaction in chimpanzee. *Journal of Comparative and Physiological Psychology*, 44(1), 9.

O'Mara, H. (1991). Quantitative and methodological aspects of stimulus equivalence. *Journal of the experimental analysis of behavior*, 55(1), 125–132.

O'Reilly, R. C. & Munakata, Y. (2000). *Computational explorations in cognitive neuroscience: Understanding the mind by simulating the brain*. MIT press.

Ortega, D. & Lovett, S. (2018). Equivalence-based instruction to establish a textual activity schedule in an adult with down syndrome. *Behavioral Interventions*, 33(3), 306–312.

Paparo, G. D., Dunjko, V., Makmal, A., Martin-Delgado, M. A., & Briegel, H. J. (2014). Quantum speedup for active learning agents. *Physical Review X*, 4(3), 031002.

Piaget, J., Chilton, P. A., & Inhelder, B. (1971). *Mental imagery in the child: a study of the development of imaginal representation.* London : Routledge & Kegan Paul.

Placeres, V. (2014). An analysis of compound stimuli and stimulus equivalence in the acquisition of russian vocabulary. Master's thesis, Youngstown State University.

Rapanelli, M., Frick, L. R., Fernández, A. M. M., & Zanutto, B. S. (2015). Dopamine bioavailability in the mpfc modulates operant learning performance in rats: an experimental study with a computational interpretation. *Behavioural brain research*, 280, 92–100.

Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5), 206.

Saunders, R. R., Chaney, L., & Marquis, J. G. (2005). Equivalence class establishment with two-, three-, and four-choice matching to sample by senior citizens. *The Psychological Record*, 55(4), 539–559.

Seefeldt, D. A. (2015). *Evaluation of Equivalence Relations: Models of Assessment and Best Practice*. PhD thesis, Southern Illinois University.

Sidman, M. (1971). Reading and auditory-visual equivalences. *Journal of Speech, Language, and Hearing Research*, 14(1), 5–13.

Sidman, M. (1990). Equivalence relations: Where do they come from? *Behaviour analysis in theory and practice: Contributions and controversies*, (pp. 93–114).

Sidman, M. (1994). *Equivalence relations and behavior: A research story.* Authors Cooperative.

Sidman, M. (2009). Equivalence relations and behavior: An introductory tutorial. *The Analysis of verbal behavior*, 25(1), 5–17.

Sidman, M. (2013). Techniques for describing and measuring behavioral changes in alzheimers patients. *European Journal of Behavior Analysis*, 14(1), 141–149.

Sidman, M., Cresson Jr, O., & Willson-Morris, M. (1974). Acquisition of matching to sample via mediated transfer 1. *Journal of the Experimental Analysis of Behavior*, 22(2), 261–273.

Sidman, M., Rauzin, R., Lazar, R., Cunningham, S., Tailby, W., & Carrigan, P. (1982). A search for symmetry in the conditional discriminations of rhesus monkeys, baboons, and children. *Journal of the experimental analysis of behavior*, 37(1), 23–44.

Sidman, M. & Tailby, W. (1982). Conditional discrimination vs. matching to sample: An expansion of the testing paradigm. *Journal of the Experimental Analysis of behavior*, 37(1), 5–22.

Sidman, M., Willson-Morris, M., & Kirk, B. (1986). Matching-to-sample procedures and the development of equivalence relations: The role of naming. *Analysis and intervention in Developmental Disabilities*, 6(1-2), 1–19.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. (2016). Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587), 484.

Spencer, T. J. & Chase, P. N. (1996). Speed analyses of stimulus equivalence. *Journal of the Experimental Analysis of Behavior*, 65(3), 643–659.

Staddon, J. & Bueno, J. L. O. (1991). On models, behaviorism and the neural basis of learning. *Psychological Science*, 2(1), 3–11.

Steele, D. & Hayes, S. C. (1991). Stimulus equivalence and arbitrarily applicable relational responding. *Journal of the Experimental Analysis of Behavior*, 56(3), 519–555.

Steingrimsdottir, H. S. & Arntzen, E. (2011). Using conditional discrimination procedures to study remembering in an alzheimer's patient. *Behavioral Interventions*, 26(3), 179–192.

Sutton, R. S. & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

Tolman, E. C. (1948). Cognitive maps in rats and men. *Psychological review*, 55(4), 189.

Toussaint, K. A. & Tiger, J. H. (2010). Teaching early braille literacy skills within a stimulus equivalence paradigm to children with degenerative visual impairments. *Journal of Applied Behavior Analysis*, 43(2), 181–194.

Tovar, A. E. & Chávez, A. T. (2012). A connectionist model of stimulus class formation with a yes/no procedure and compound stimuli. *The Psychological Record*, 62(4), 747–762.

Tovar, Á. E. & Westermann, G. (2017). A neurocomputational approach to trained and transitive relations in equivalence classes. *Frontiers in psychology*, 8, 1848.

Vernucio, R. R. & Debert, P. (2016). Computational simulation of equivalence class formation using the go/no-go procedure with compound stimuli. *The Psychological Record*, 66(3), 439–449.

Walker, B. D., Rehfeldt, R. A., & Ninness, C. (2010). Using the stimulus equivalence paradigm to teach course material in an undergraduate rehabilitation course. *Journal of Applied Behavior Analysis*, 43(4), 615–633.

Wang, C.-C., Kulkarni, S. R., & Poor, H. V. (2005). Bandit problems with side observations. *IEEE Transactions on Automatic Control*, 50(3), 338–355.

Zhang, Z., Beck, M. W., Winkler, D. A., Huang, B., Sibanda, W., Goyal, H., et al. (2018). Opening the black box of neural networks: methods for interpreting neural network models in clinical applications. *Annals of translational medicine*, 6(11).

A A Detailed Example on How the Model Works

In the following, we explain an experiment through modeling the Protocol 1. In the beginning, one of A_1, A_2, and A_3 is chosen with probability $P^{(t)}(s) = 1/3$ to be shown as the sample stimulus, where the comparison stimuli (or actions) will be B_1, B_2, and B_3. Hereafter, for simplicity, we use the same notations for actual stimuli and the

remembered clips of the stimuli, say $A_1 = I(A_1)$, unless there is an ambiguity. In the Figures 11-15, the inside of the rectangle shows the agent memory (clip network) while the outside shows the environment and actual stimuli. Moreover, red color is used for the sample stimuli and its internal clip at current trial, while blue color is used for the comparison stimuli at the same trial. Solid links are used for baseline/direct relations and dashed links represent symmetry links.

Consider at time $t = 1$, sample stimuli A_1 is presented to the agent. So A_1 is added to the percept set; i.e. $S = S \cup \{A_1\}$. Also, a memory clip representing A_1 is created and added to the memory space; $C = C \cup \{A_1\} = \{A_1\}$.

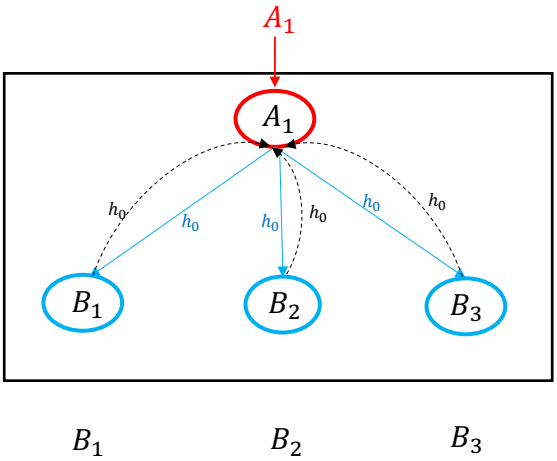
Based on the learning protocol, the set of comparison stimuli B_1, B_2, B_3 will appear after 1s delay.[20] Then, three memory clips for B_1, B_2, and B_3 are created by the agent and added to the C space. The actuator space has now three members as well, so $A = A \cup A_1 = \{B_1, B_2, B_3\}$.

The new connections and h-values must be initiated, since this sample-stimulus/comparison-stimuli is presented for the first time. At this stage, six edges will be established which their initial h-values are h_0; i.e. $h^{(1)}(A_1, B_1) = h^{(1)}(A_1, B_2) = h^{(1)}(A_1, B_3) = h_0$, and $h^{(1)}(B_1, A_1) = h^{(1)}(B_2, A_1) = h^{(1)}(B_3, A_1) = h_0$. As a result, the conditional probability distribution $\{p^{(1)}(a|s)\}_{A_1}$ is uniform for all possible actions in the memory space, see Figure 11a.

Consider that the agent chooses $a^{(1)}$ where:

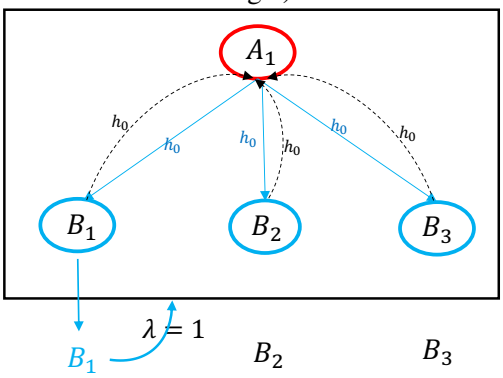
1. $a^{(1)} = B_1$, i.e. the agent chooses the correct option which must be reinforced by $= 1$. In this case, $h^{(2)}(A_1, B_1)$ will be increased by K_1 due to Eq. (4). K_1 is set to unitary based on PS.
   Moreover, we expect that strengthening A_1B_1, affects the formation of B_1A_1 (symmetry relation in SE), so $h^{(2)}(B_1, A_1) = h_0 + K_2$. Other transitions remain unchanged; i.e. equal to h_0; see Figure 11b and 11c.
2. $a^{(1)} = B_2$, i.e. the agent chooses a wrong option (exactly the same for $a^{(1)} = B_3$ at this stage), so $= 1$. This negative reward reinforces other options, but the negatively rewarded one, see Figure 11d and 11e.
   In this example, the transition weight from clip A_1 to clips B_1 and B_3 will be increased by K_3 where $K_3 \quad \frac{K_1}{2}$; see Eq. (6). The symmetry updates are similar, i.e. the transition weight from clip B_2 to clip A_1 will not change, and the transition weights from clips B_1 and B_3 to clip A_1 will be increased by an additive factor K_4 where $0 < K_4 \quad \frac{K_2}{2}$; due to Eq. (7).

[20] Please note that, when the relation A_1B_1 is the desired relation to be reinforced, the comparison stimuli are chosen from B category. The number of them could be different but at least two. In this case, each category contains three members and so we just have one option of 3-member comparison, where the location of shown stimuli does not take into account. But, if there are more members in categories, we have more options.



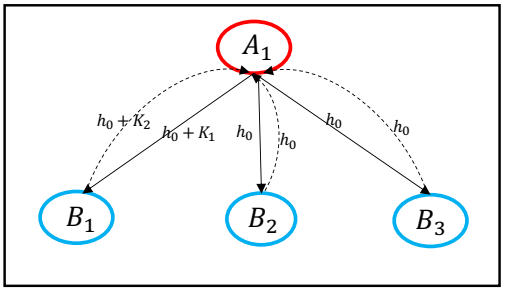
B_1 B_2 B_3

(a) The first stimulus sample A_1, followed by the three comparison samples $B_1; B_2; B_3$ (outside the rectangle). The clips are added to the memory, and initialized with h_0 (inside the rectangle).

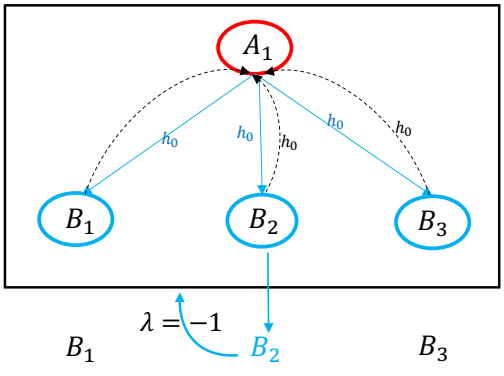


B_1 B_2 B_3

(b) The correct pair is chosen, so agent receives positive reward $= 1$.

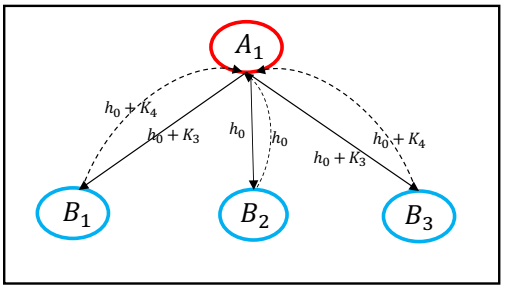


(c) The h-value for A_1B_1 connection will be added by K_1 and as a symmetry effect the h-value of B_1A_1 will be added by K_2, where K_2 K_1.



B_1 B_2 B_3

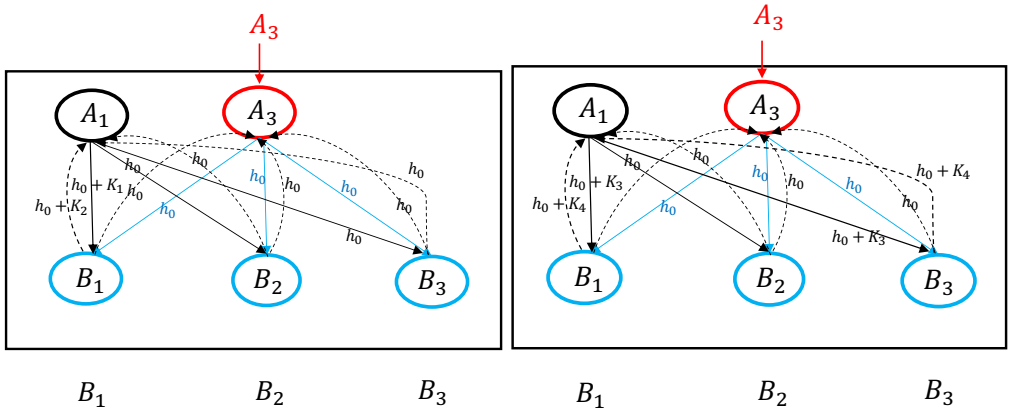
(d) If a wrong option is chosen, negative reward $=$ 1 will be return to the agent.



(e) Negative reward, will amplify other options. So h-values for A_1B_1 and A_1B_3 will be added by K_3. h-values for symmetry connection, i.e. B_1A_1 and B_3A_1 will be added by K_4.

Figure 11: The first trial for A_1B_1 through positive and negative rewards at time step $t = 1$. The agent creates clips for all the perceived stimuli (11a) and updates the connection weights based on environment feedback. The updating rule in positive (11b, 11c), and negative (11d, 11e), is presented. The percept clip (sample stimulus) is shown in red, and the action clips (comparison stimuli) are represented in blue.

Let $t = 2$, and the sample stimulus to be A_3, while the comparison stimuli are the same as previous time step i.e. $A_2 = A_1$; so no new action is added into the action space A. A_3 is added to the percept space now $S = S \cup \{A_3\} = \{A_1; A_3\}$. Note that percept and action spaces are not shown in the Figures and that we only depict how an agent updates its memory clips during training. Since the trial setting is new, all the transitions will be established, initialized and updated like the previous time step. This is similar to the first time that A_2B_2 pair is supposed to be learned. See the clip network C (inside the rectangle) in Figure 12a and Figure 12b when clip A_3 is added.



(a) A new sample stimulus A_3 is presented by environment, and an agent subsequently creates a new clip and initializes its connections to the action clips with h_0. This network is based on a correct choice in the first trial (Figure 11b).

(b) A new sample stimulus A_3 presented by environment, and an agent subsequently creates a new clip and initializes its connections to the action clips with h_0. This network is based on a wrong choice in the first trial (Figure 11d).

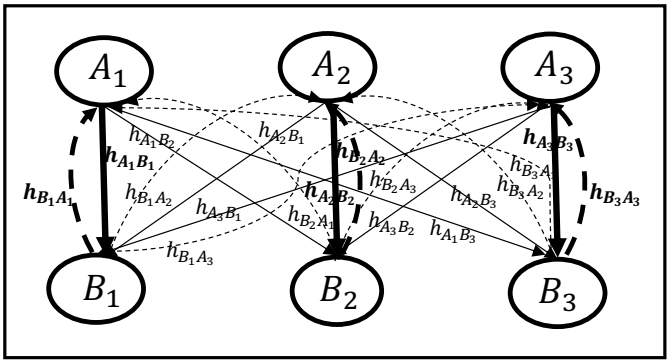
Figure 12: The second trial ($t = 2$) where A_3 is the sample stimulus. The agent creates a new clip for A_3 and updates the h-values based on the learning history; i.e. if the network has been updated with a chosen correct pair (Figure 11c) or with a chosen wrong pair (Figure 11c). Only h-values between the current sample stimulus (A_3 in red) and comparison stimuli ($B_1; B_2; B_3$ in blue) will be updated at this trial.

Now consider that experiment repeated the trials until all the desired AB relations are trained and 90% of agent's choices within the last 30 trials be correct, see Figure 13a for a schematic representation. The thick solid links between A_1B_1, A_2B_2, and A_3B_3 show the mastery of these baseline relations. The thick dashed links show symmetry formation. The weak links illustrate that although the agent is well trained, there is still a low chance of wrong choice in a MTS trial. Based on Protocol 1, the environment trains BC relation next.

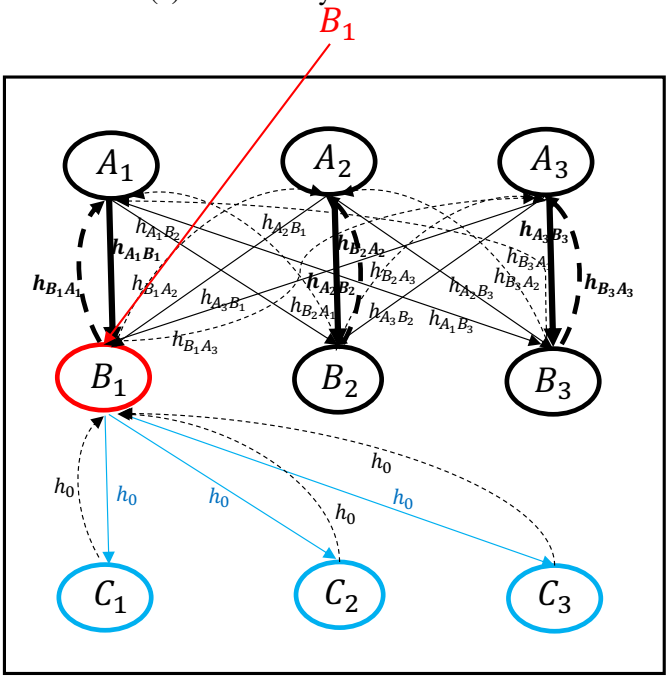
- At time t^0, let the sample stimulus be B_1 and the comparison stimuli be $A_{t^0} = \{C_1; C_2; C_3\}$ (Figure 13b, outside the rectangle). At this point, the percept space is $S = S \cup \{B_1\} = \{A_1; A_2; A_3; B_1\}$. Similarly, the action space would be $A = A \cup A_{t^0} = \{B_1; B_2; B_3; C_1; C_2; C_3\}$, and the clip space would be $C = \{A_1; A_2; A_3; B_1; B_2; B_3; C_1; C_2; C_3\}$, for clip space representation, see Figure 13b, inside the rectangle.

Note that clip B_1 in the agent memory both represents a percept clip and an action clip.

– At time t^0, three input and three output links will be established from B_1 and initialized with $h_0 = 1$, see Figure 13b. The probabilities for all comparison stimuli are then uniform $p^{(t^0)}(B_1 j C_1) = p^{(t^0)}(B_1 j C_2) = p^{(t^0)}(B_1 j C_3) = 1=3$. Similar to the AB training step, the environment reinforces the desired relation and by accomplishing this training phase, we expect a network like the one presented in Figure 14a. Thick solid links show the well trained baseline relations, thick dashed relations represent formation of symmetry relations, and weak links show a weak possibility for wrong option in MTS trials.



(a) The mastery of AB relation.



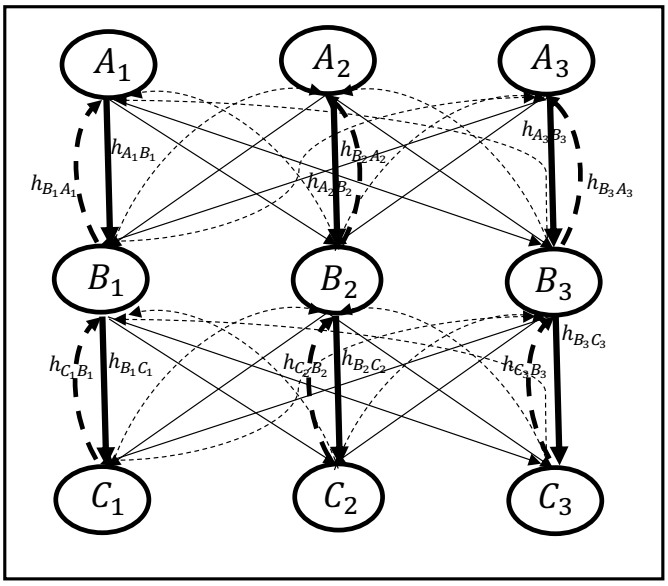
(b) B_1 as sample stimulus (outside the rectangle) activate the existed clip B_1 (in the rectangle, in red). C_1, C_2, and C_3 are the comparison stimuli (outside the rectangle). Agent creates new clips for them and initializes the links between percept clip B_1 and action clips C_1, C_2, and C_3 (inside the rectangle).

Figure 13: When AB relation is trained 13a, and B category members appear as the sample stimulus, clips in B category will be activated as the percept clips 13b and C_1, C_2, and C_3 will be action clips.

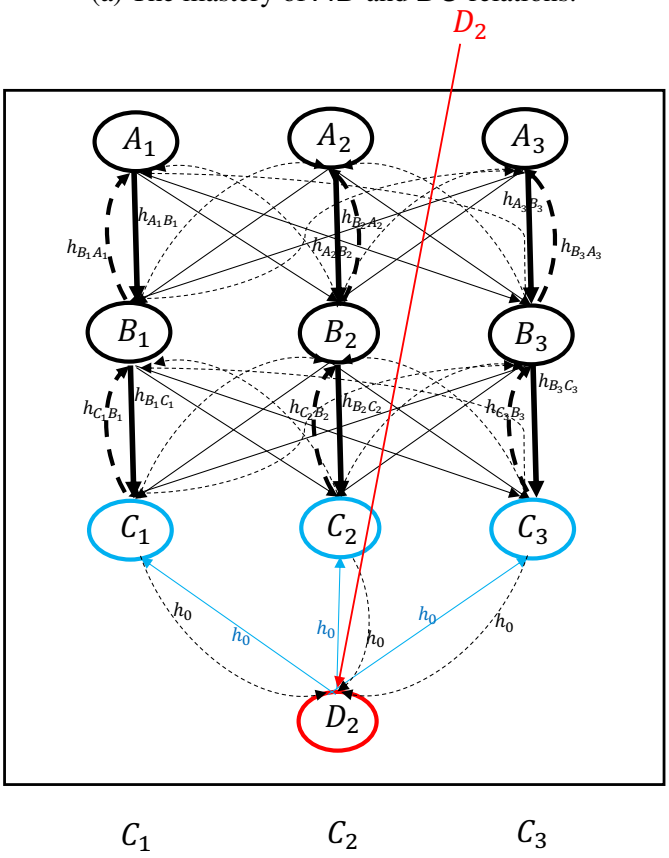
– Suppose that BC relation is also trained and passed the criterion (Figure 14a, thick connections). The final step in training phase is DC relation.

  Let D_2 be the sample stimulus at time t^{00}, and $A_{t^{00}} = fC_1; C_2; C_3g$ (Figure 14b, outside the rectangle). So, D_2 will be added to S, but the action space does not change. A clip for D_2 would be added to C and the initial links will be established and initialized with h_0 (Figure 14b,

inside the rectangle). The first choice is uniformly selected with probability 1=3, but after enough MTS trials, the probability of desired pair meets the criterion.



(a) The mastery of AB and BC relations.



C_1 C_2 C_3

(b) D_2 as sample stimulus (outside the rectangle) makes the agent to create a new clip D_2 (in the rectangle, in red). C_1, C_2, and C_3 are the comparison stimuli (outside the rectangle). Agent does not create new clips for them, but only initializes the links between new clip D_2 and existed clips C_1, C_2, and C_3 (inside the rectangle).

Figure 14: When AB and BC relations are trained 14a, and training the relations DC is the next step. In 14b, D_2 appears as the sample stimulus and its connections are initialized to the C category which plays the action set role.

Figure 15 shows the memory network after a successful training phase, where thick connections are the trained relations and weak connections show the wrong unfavourable possible choices. For the testing phase, we can compute the agent's policy and see if the conditional probability for symmetry, transitivity, and equivalence relations are according to the protocol. If the desired one passes the criterion, we will say that the equivalence relations are formed for the agent. In the simulation part, section 4, we address the testing phase via different methods to compute probability distribution for action set when the relation is a derived one. We replicate the testing phase similar to real experiments along with computation of probabilities.

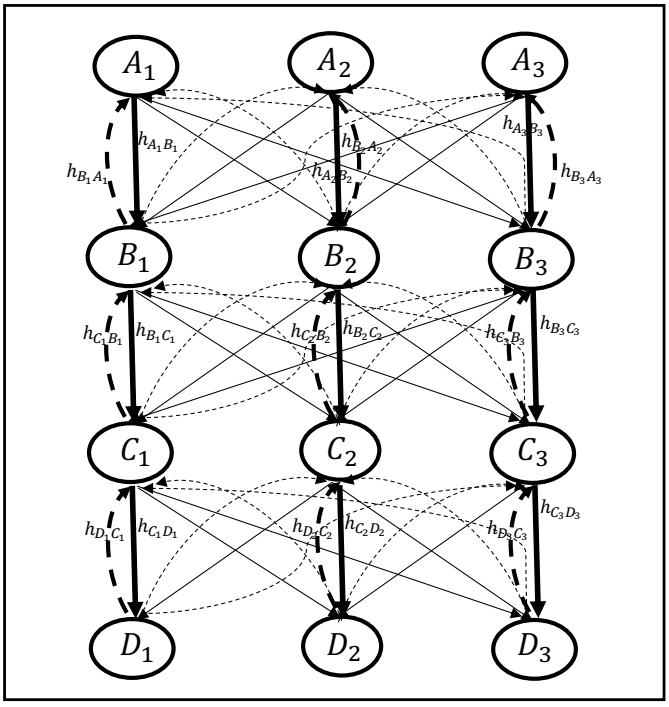


Figure 15: A representation of the memory clip network after training phase. We show the symmetry connections with dash-line in order to clarify that they are not reinforced directly during MTS procedure.

B Calculation of probability distribution over an action set with max-product

Computation of probabilities from h-values is an important phase, since the agent updates h-values but actions are taken based on probabilities. Two general methods which are used in original PS, and similarly in EPS, are called 'standard' model and 'softmax' model; they respectively use simple normalisation and softmax function over h-values. However, this could be more challenging when the direct connections do not exist. In this case, one might consider other conditions that might change the computed probabilities. In the following we explain a few possibilities for computing probabilities in max-product scenario, in which we addressed in Eq.(8) and section 4.1. In Figure 16, a

sample structure of the agent's memory clip is presented where the h-values are positive and probabilities during training are computed by normalization of h-values.

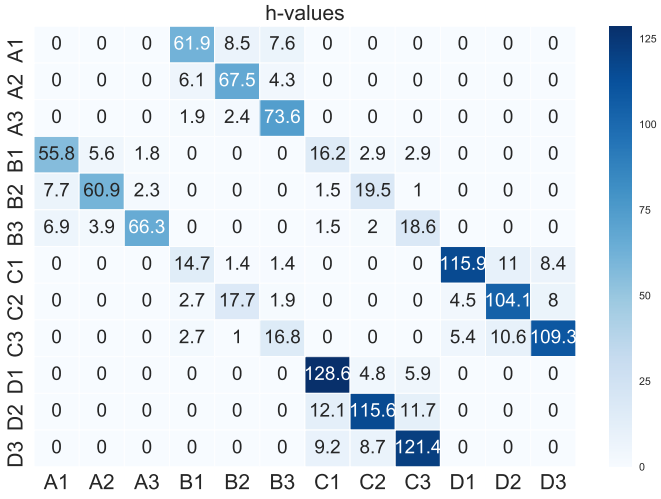
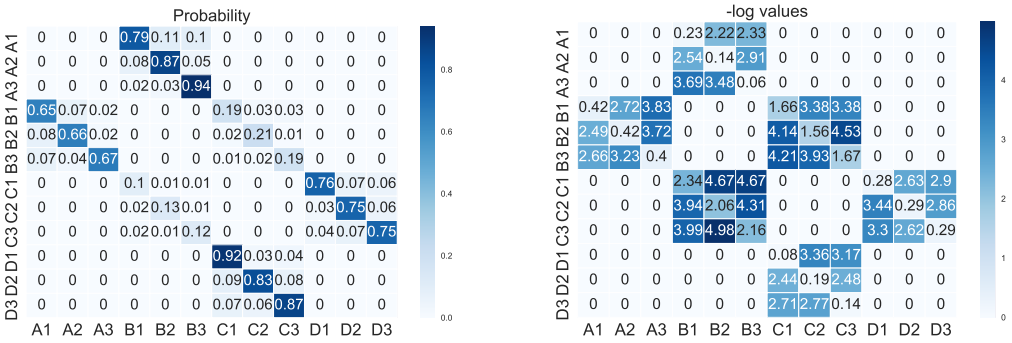


Figure 16: A sample configuration of network h-values after training AB, BC and DC based on protocol 1 when $= 0{:}0001$, $K_1 = 1$, $K_2 = 0{:}9$, $K_3 = 0{:}5$, $K_4 = 0{:}45$.

In Figure 16, we see that the range of h-values for different categories could be quite diverse. For instance h-values between stimuli in category D and C is about 6 times bigger than h-values between stimuli in category B and C. This means that the agent is selected more efficiently in BC training trials and pass the criterion more quickly, but behaves less efficiently in DC training trials and needs more blocks of training to meet the criterion. This will affect the probabilities as represented in Figure 17.



(a) Transition probability of network in Figure 16, using normalization.

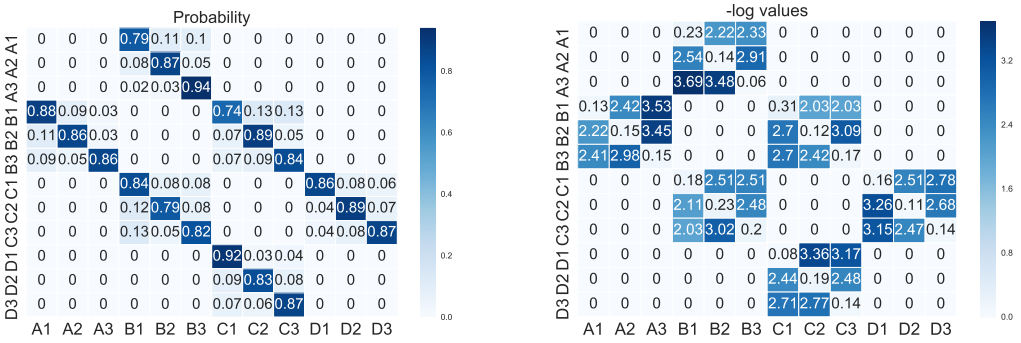
(b) Negative log values of the probabilities to convert max-product problem into a min-sum problem.

Figure 17: Transition probabilities and negative log of probabilities of the sample network in Figure 16.

Suppose that the testing trial has A_2 as the sample stimulus and $C_3; C_1; C_2$ as the action stimuli. As reported in Table 8, the path with lowest cost could pass through

a category more than once, say $A_2; B_2; C_2; D_2; C_3$. Note that the reported simulation results in the paper, as we refereed to as Dijkstra's algorithm, is similar to Figure 17, i.e. without any extra conditions.

One might argue that the probabilities must be marginalized based on the categories. In other words, the agent first targets a specific category, then at the second level, chooses a member of that category. Therefore, the probability must be normalized for each category. In Figure 18, a category-based computation in which probabilities are marginalized is presented.



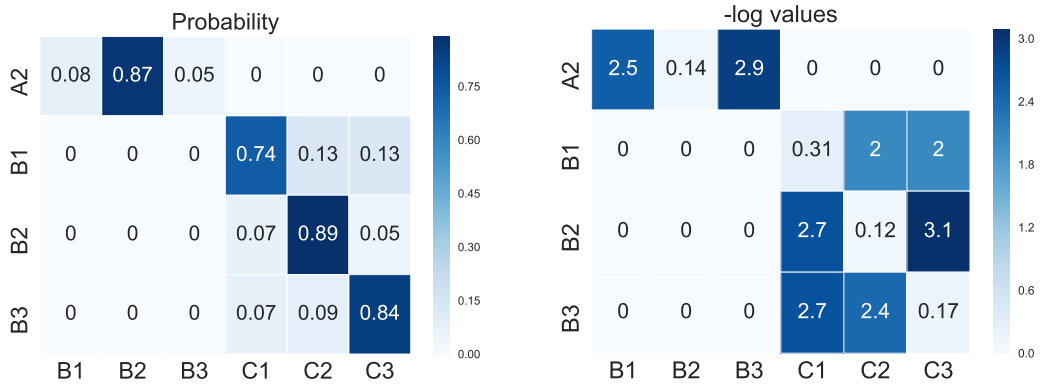
(a) Transition probability of network in Figure 16, normalization based on category.

(b) Negative log values of the probabilities to convert max-product problem into a min-sum problem.

Figure 18: Transition probabilities and negative log of probabilities of the sample network in Figure 16 when category is taken into account.

From Table 8 we observe that the calculated probability vector of category-based computation is higher than the previous case in general, but comparison of the normalized vector shows the probability of correct choice, A_2C_2 in category-based computation is slightly less than its counterpart. The explanation is that in category-based version, a multiplicative factor which represents the probability of choosing each different category is removed. This affects the probabilities and therefore produces different final distributions. Consider that, if the h-values for different categories are in the same range, which is what we expect, this multiplicative factor would be the same for all the actions and the normalized probabilities will be the same. Remember that in the category-based calculation of probabilities, the lowest cost path could pass through a category more than once similar to the first case.

The third scenario which we refer to as Viterbi algorithm, avoids passing a category more than once and is based on a trellis diagram from the network. The trellis diagram is an ordered graph from a start point to the destination layer. The trellis diagram for EPS is configured for each trial and has c_s as the start point, the layers consist of members of passing categories and the destination layer is A_t. The strategy to find the passing categories from c_s to c_a is simply by finding the shortest path from c_s to c_a, keep the members of categories with at least a member is in the path, and remove other nodes and edges which have the opposite direction. Figure 19 shows the probabilities on the trellis diagram and negative log values.



(a) Transition probability of network in Figure 16, normalization based on trellis diagram for percept and actions.

(b) Negative log values of the probabilities to convert max-product problem into a min-sum problem.

Figure 19: Transition probabilities and negative log of probabilities of the sample network in Figure 16 when a trellis diagram based on the trial is made first, before computing the probabilities.

Table 8: Details of computing derived probabilities form sample clip in Figure 16.

Computation method	A_2C_3	A_2C_1	A_2C_2
No condition (Figure 17)			
Lowest cost path	$A_2; B_2; C_2; D_2; C_3$	$A_2; B_1; C_1$	$A_2; B_2; C_2$
min-sum value	4:4697	4:2016	1:7019
Calculated probability	0:0115	0:015	0:1823
Normalized probability	0:0549	0:0717	0:8734
h-values	1:0	1:3075	15:9229
Category-based (Figure 18)			
Lowest cost path	$A_2; B_2; C_2; D_2; C_3$	$A_2; B_2; C_2; B_1; C_1$	$A_2; B_2; C_2$
min-sum value	2:8554	2:6740	0:2625
Calculated probability	0:0575	0:069	0:7692
Normalized probability	0:0642	0:0770	0:8588
h-values	1:0	1:1989	13:3693
Viterbi (Figure 19)			
Lowest cost path	$A_2; B_3; C_3$	$A_2; B_2; C_1$	$A_2; B_2; C_2$
min-sum value	3:0743	2:8471	0:2625
Calculated probability	0:0462	0:0580	0:7692
Normalized probability	0:0529	0:0664	0:8807
h-values	1:0	1:2551	16:6406

Probability of correct match, A_2C_2 from Viterbi scenario is slightly higher than the previous methods; the explanation is that by removing some edges and forbidding to pass through a category twice, the lowest cost path in wrong options might be removed.

After finding the probabilities for all the possible actions $a \ 2 \ A_t$, we can compute h-values of the connections using Eq.(11).

$$h^{(t)}(c_s; c_a) = \frac{p^{(t)}(c_a|c_s)}{p_{\min}} h_0 \tag{11}$$

where $p_{\min}$ is the minimum of achieved probability where we set its h-value equal to h_0. Note that if we use softmax function to compute probabilities, converting probabilities to the h-values is through Eq.(12).

$$\mathsf{h}^{(t)}(c_s; A_t) = \frac{1}{-} \ \log(p^{(t)}(c_{a_1}|c_s)) \qquad \log(p^{(t)}(c_{a_m}|c_s)) \qquad [\mathsf{h}_{\min} + h_0; \qquad \mathsf{h}_{\min} + h_0] \tag{12}$$

where $\mathsf{h}_{\min}$ is the minimum value of the computed h-values which is used to put the minimum value of h-values to h_0.

It is worth mentioning that the final results in max-product scenario, in spite of the chosen strategy for calculation of probabilities, are quite similar. However, the selected method affects the interpretation of the mechanism of agent's memory in order to make a decision on a derived relation, which might be of interest in EPS model.