Child-friendly divorcing: Incremental Hierarchy Learning in Bayesian Networks

Florian Röhrbein, Julian Eggert, Edgar Körner

2009

Preprint:

This is an accepted article published in Proceedings of the 2009 International Joint Conference on Neural Networks (IJCNN). The final authenticated version is available online at: https://doi.org/[DOI not available]

Röhrbein, F., Eggert, J., Körner, E. (2009). Child-friendly divorcing: Incremental hierarchy learning in Bayesian Networks, *Proceedings of International Joint Conference on Neural Networks* (IJCNN), 2711-2716.

Honda Research Institute Europe GmbH
http://www.honda-ri.de/
Carl-Legien-Strasse 30
63073 Offenbach am Main
Germany

Child-Friendly Divorcing: Incremental Hierarchy Learning in Bayesian Networks

Florian Röhrbein, Julian Eggert, and Edgar Körner

Abstract—**The autonomous learning of concept hierarchies is still a matter of research. Here we present a learning schema for Bayesian networks which results in a nested structure of sub- and superclass relationships. It is based on so-called parent divorcing but exploits the similarity of all nodes involved as expressed by their connectivity pattern. If the procedure is applied to simple object-property pairings a nested taxonomic hierarchy emerges. We further show how the learning procedure can be aligned with basic results from developmental psychology. For this we made a set of simulations which clearly indicate that a fixed developmental order of sensory maturation is crucial for the emerging conceptual system. The learning procedure itself is biologically plausible since it works incrementally, makes use of only local information and leads to a reduced computational effort by building a more efficient representation.**

I. Introduction

THERE are huge collections of textual knowledge available which provide information about the properties of objects and the relations between them (e.g. the Open Mind Common Sense database OMCS, [6]). Of special interest for a Bayesian treatment are relations like *has component*, *has part, comprises, includes, has property* etc. because they can be given a "causal" interpretation in the following manner: locations (e.g. a *kitchen*) "cause" an observer to see objects which typically are located herein (e.g. a *cup*) with a certain probability, these objects "cause" the observer to see the parts they usually are made of (e.g. a *handle*) and the parts in turn "cause" to see certain colors (e.g. *white*) and shapes (e.g. *curved*). The corresponding knowledge snippets from the database (here *cup is located in kitchen, cup has part handle* etc.) can be used to built up a standard Bayesian network (BN) in which each node represents a concept (*kitchen, cup* etc.) and each link represents a causal relation. In BNs conditional probability tables (CPTs) are attached to each node and these can be instantiated with confidence scores which are often provided alongside the knowledge snippets (leading thus to conditional probabilities like p(*cup*|*kitchen*)). In [8] we made use of such scored assertions and combined them with several sources to build large probabilistic networks.

A problem that immediately arises if it comes to Bayesian reasoning is that the number n of parent nodes becomes critical since the CPT entries grow exponentially with n. If we consider objects and their properties this becomes especially pressing since simple features are observed in a huge number of objects. One well-known solution to this complexity issue is to use additional nodes by standard parent divorcing, a procedure introduced by [5]. Employing this procedure results in an addition of "divorcing nodes" which leads to an arbitrary grouping of the corresponding child nodes. At this point we asked ourselves how this procedure can be advanced in a way that it leads to new nodes which are no more meaningless, but can be interpreted as subclasses, interconnecting appropriate properties with corresponding objects. We came up with a procedure called "child-friendly divorcing" which is the topic of this contribution.

The paper is organized as follows: First the method is introduced, starting with standard divorcing and advancing to the modified and extended version. We illustrate the behavior with small-scale examples and show results in terms of coding gain and learned network structure. The learning procedure then is employed in an incremental setting that interestingly results in network structures that depend on the sequence of incoming information. We give a report on these results and relate them in the next section to empirical findings from developmental psychology. Finally we discuss relations to other work.

Manuscript received January 5, 2009.

Florian Röhrbein was with the Honda Research Institute Europe GmbH, Carl-Legien-Str. 30, 63073 Offenbach am Main, Germany. He is now with the University of Bremen, Enrique-Schmidt-Str. 5, 28359 Bremen, Germany (phone: +49-421-218-64231, fax: +49-421-218-64239, e-mail: roehrbei@informatik.uni-bremen.de).

Julian Eggert and Edgar Körner are with the Honda Research Institute Europe GmbH (e-mail: julian.eggert@honda-ri.de, edgar.koerner@honda-ri.de).

II. Learning Procedure

A. Standard Parent Divorcing

We base our learning procedure on a technique used previously merely as a design tool for building more efficient Bayesian models. The basic idea of so-called parent divorcing (see [5]) is to split parent nodes by introducing an intermediate node which leads to an increased computational efficiency due to less entries in the CPTs.

Fig. 1 illustrates the schema with nodes representing variables O_i (coding for a set of objects) plus one node representing property p_1 which is common to these objects. Divorcing amounts in separating parents $O_1,...,O_k$ from parents $O_{k+1},...,O_m$ by introducing a mediating variable X, thus making X a parent of p_1 and a child of $O_{k+1},...,O_m$.

The underlying assumption is that the set of configurations $(O_{k+1},...,O_m)$ can be portioned into two sets such that whenever two configurations $(o'_{k+1},...,o'_m)$ and $(o''_{k+1},...,o''_m)$ are elements of the same set (i.e. the

same state of X), then $p(p_1 \mid O_1,...,O_k,o'_{k+1},...,o'_m)$ equals $p(p_1 \mid O_1,...,O_k,o''_{k+1},...,o''_m)$.

Using parent divorcing in conjunction with independence of causal influence may reduce the complexity of inference exponentially. But it is important to note that the procedure can always be applied: If mediating variable X has one state for each configuration of its parents, then $p(p_1 \mid o_1,...,o_m)$ is equivalent to $p(p_1 \mid x,o_1,...,o_k)$. In this extreme case nothing has been gained with respect to reducing the model's complexity.

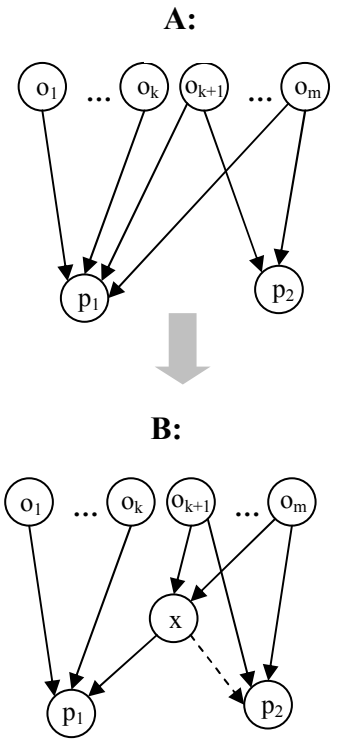

Fig. 1. Standard divorcing applied to network A leads to network B with mediating variable X. The gain in terms of CPT entries depends on the number of states needed for X to keep the joint probability distribution unchanged.

B. Child-friendly Parent Divorcing

The base algorithm is now modified to avoid a splitting of parent nodes into arbitrary groups but instead group together parents that are similar in some respect. The rationale is that caring for similarity might foster the emergence of meaningful groupings represented by the new mediating nodes. Node similarity boils down to similarity in connectivity, and here we just look at the number of common child nodes. The relation to these child nodes is made explicit by the insertion of additional links. The algorithm work as follows:

1) Generating of and Linking to New Nodes

In order to trigger divorcing we first have to check if for some node the number of parents is above a certain threshold. If this is the case (as for node p_1 in Fig. 1), we determine which additional child nodes (e.g. p_2) the to-be-divorced parent nodes have in common and partition the parents into two sets with respect to their overlap in child nodes. In the unlikely case that the node which triggered the divorcing is the only common child we decide not to divorce at all.

2) Adding Non-Essential Links

After having inserted the mediating node and having changed the connections appropriately we add non-essential links from this newly generated node to all nodes of the set of common parent nodes, for which the following criterion is met: In order to avoid redundancy, there should not be a path between mediating node and node in question. For the example in Fig. 1 this simply amounts to inserting a link from x to p_2 (dashed arrow).

3) Handling of Already Learned Nodes

Finally, we have to deal with the case that the connectivity pattern of the to-be-established node would be the same as for an already existing node (which has been learned previously). If this is the case we dispense with the new node and use the existing one instead in the following way: The connection from the to-be-divorced parent node to the child node that triggered divorcing is to be replaced by a connection from father node to already learned mediating node. E.g., in the configuration of Fig. 1B assume we have added knowledge about an object O_{m+1} which has properties p_1 and p_2. "Child-friendly parent divorcing" would select O_{m+1} and x for divorcing, since they have children p_1 and p_2 in common. A new mediating node would result in a node with the same child nodes (p_1 and p_2), therefore $O_{m+1} \rightarrow p_1$ is replaced by $O_{m+1} \rightarrow x$.

C. Correctness

An intuitive account on the correctness of the resulting network modifications is as follows: Parent divorcing is known to leave the joint probability distribution unchanged and in (1) we just constrained the cases in which we like to divorce. Adding links (2) is completely uncritical, at worst the network complexity will be increased. We have to consider the special case (3) in more detail.

For the network fragment of Fig. 1 we have two joint probabilities before (A) and after (B) divorcing:

$$p_A(p_1,o_1,...,o_m) = p(p_1 \mid o_1,...,o_m) \cdot p(o_1) \cdot ... \cdot p(o_m)$$

$$\begin{aligned} p_B(p_1,o_1,...,o_m) &= \sum_x p_B(p_1,x,o_1,...,o_m) \\ &= p(o_1) \cdot ... \cdot p(o_m) \cdot \\ &\quad \sum_x p(p_1 \mid x,o_1,...,o_k) \cdot p(x \mid o_{k+1},...,o_m) \end{aligned}$$

Since the transformation should leave the joint probability distribution unchanged, $p_A(p_1,o_1,...,o_m)$ has to equal

$p_B(p_1, o_1,...,o_m)$. In order to fulfill this equality, we have to find conditional probabilities in B which fit to given conditional probabilities in A such that

$$p(p_1 \mid o_1,...,o_m) = \sum_x p(p_1 \mid x, o_1,...,o_k) \cdot p(x \mid o_{k+1},...,o_m)$$

This equality also holds for additional child nodes p_i since they cancel out on both sides. There are $2^{k+2} + 2^{m-k}$ conditional probabilities in B which have to be adjusted according to the 2^m values in A. It is easy to show that the number of values in B is always equal or less than the number of values in A if there is more than one parent node to node x.

As far as the number of conditional probabilities is concerned, parent divorcing with binary variables thus always leads to a coding gain for reasonable node configurations.

We now restrict the choice of parameters further by setting $p(p_1 = TRUE \mid x = TRUE, o_1,...,o_k) = 1$, which will be motivated below. With this constraint the equality reads

$$p(p_1 \mid o_1,...,o_m) = p(x = TRUE \mid o_{k+1},...,o_m) + \\ + p(p_1 \mid x = FALSE, o_1,...,o_k) \cdot p(x = FALSE \mid o_{k+1},...,o_m)$$

This condition has to be met for all possible values of $p_1, o_1,...,o_m$ and we reformulate it to the following expression which allows for an easily interpretation:

$$p(p_1 \mid o_1,...,o_m) = \\ 1 - [1 - p(p_1 \mid x = FALSE, o_1,...,o_k)] \cdot \\ \cdot [1 - p(x = TRUE \mid o_{k+1},...,o_m)]$$

The joint probability thus equals a probability summation of two independent causes, since the factor that prevents $p(p_1 \mid x = FALSE, o_1,...,o_k)$ and the factor that prevents $p(x = TRUE \mid o_{k+1},...,o_m)$ are contributing independently to the probability that $p(p_1 \mid o_1,...,o_m)$ is prevented.

The trick that leads to this equation was the fixing of the conditional probability of p_1 to 1 if the new variable x has value TRUE. This in turn is motivated by the view that the link between nodes x and p_1 should reflect a taxonomic relation with node x being a subclass of node p_1. The desired outcome, that if a subclass is found to be true, all super-classes of this node should also be true, is expressed by the constraint above.

III. Experiments

Since we are interested in a learning strategy that incrementally builds a hierarchical structure, we tested the schema described above in various settings and paid attention not only to the achieved coding gain, but also to the developing network structure. In general, there is some variation due to the random choice of the procedure in cases with more than one best parent grouping, and therefore we have to make many runs in all experimental conditions. The resulting behavior is exemplified here with a network small enough to be presentable albeit exhibiting a dense connectivity. Fig. 2 shows such a BN with 7 objects on an upper layer that are completely connected with 6 properties on the lower layer.

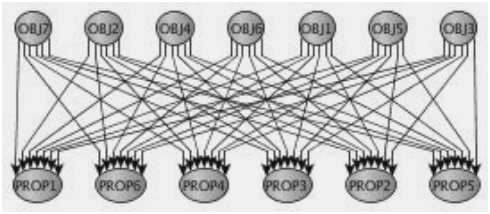

Fig. 2. Network resulting from inserting 42 object-property relations without divorcing. There are 13 nodes, linked by 42 connections which leads to 775 CPT entries.

A. Incremental Setting

We run the child-friendly procedure incrementally, i.e. after each of the 42 snippets that are used to build the BN. The node and link statistics for a total of 60 iterations are summarized in Table 1. It was disappointing to see that in many cases there is no coding gain at all, but a worsening in terms of CPT entries. Interestingly, the resulting network is highly dependent on the order in which the snippets enter the network and we consider two special conditions in the following.

	#nodes	#links	CPT entries
no divorcing	13	42	775
divorcing, by-object			
best	14	48	967
average	14.8	51.8	1840
worst	16	58	3515
divorcing, by-property			
best	22	61	459
average	18.8	53.9	702
worst	18	52	859

Table 1. Summary statistics for the example network before and after child-friendly divorcing. Best / worst cases refer to the total number of CPT entries. Two extreme cases can be distinguished: object-by-object and property-by-property sorting (see text).

B. Object-by-object Knowledge Acquisition

To characterize the effect of the ordering on the network behavior we considered two extreme cases: All relations are

entered object-by-object, i.e. all what is known about one object is entered sequentially, and after that all relations concerning a second object are entered and so on and so forth. The summary results for 30 trails can be found in Table 1 and a typical network in this condition is shown in Fig. 3. Generally, network performance decreased dramatically and the resulting network structure is relatively flat, since on average only 1.8 new mediating nodes are learned. Consequently, if the information about all objects' properties is provided at once, the system is not able to build up a hierarchical, distributed representation.

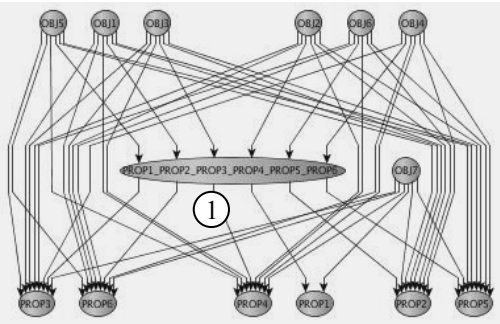

Fig. 3. Typical network resulting from child-friendly divorcing for the object-by-object condition. Only one mediating node has been learned, leaving nevertheless to 48 links and 1348 CPT entries.

C. Property-by-property Knowledge Acquisition

In an antipodal condition we fed the network with relations property-by-property, i.e. first all relations which refer to one particular property (OBJ1→PROP1, OBJ2→PROP1, …), then all relations which refer to another property etc. The resulting networks show an improvement on average (see Table 1), even though the network possesses a large number of new nodes. Fig. 4 demonstrates that these are hierarchically organized. Note that the remaining redundancy in this network is due to the symmetric start configuration.

Results in Table 1 have demonstrated a wide variety with respect to the gain in terms of CPT entries: Whereas most networks resulting from the property-by-property grouping show substantial improvements, all networks in the object-by-object condition are completely unfavorable. A further comparison reveals that neither the number of nodes nor the number of links, but the network structure is crucial (see Fig. 3, 4).

Thus the ordering seems indeed to be the critical parameter which determines the network structure. We will relate this finding to biology below and proceed with having a look at the learned nodes in the property-by-property condition. Fig. 4 shows a typical network

IV. Human Category Learning

A. Experimental Data

Similar to our simulation, also humans learn new concepts incrementally and they do this by observing and manipulating concrete instances of all concepts in question. If it comes to the development of early categories, one has additionally to take into account, that a lot of sensory information is not available from the very beginning. This is especially true for the visual system, which is very poor at birth, whereas other senses like gustation and olfaction are quite mature.

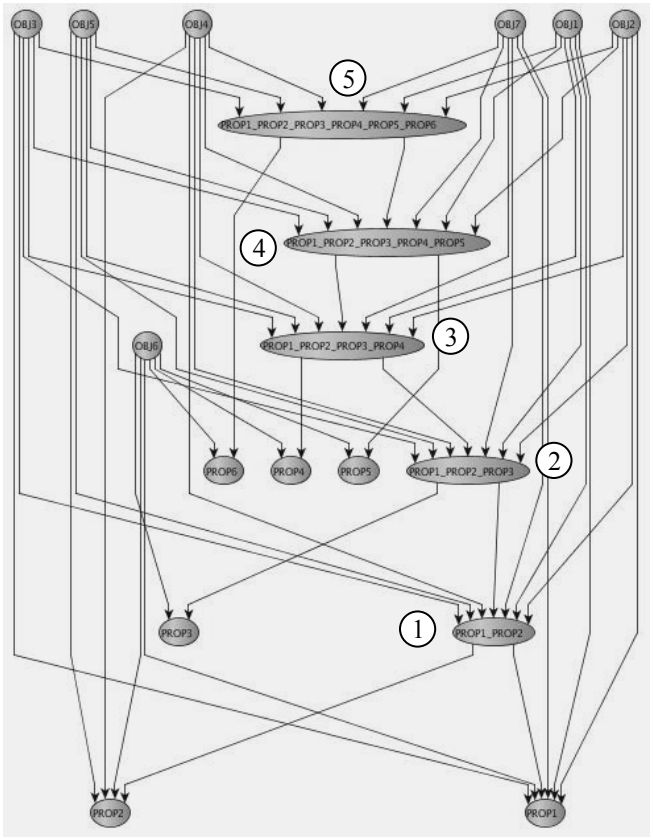

Fig. 4. Typical network resulting from child-friendly divorcing for the property-by-property condition. Here five mediating nodes have been learned, leading to a total of 52 links and 679 CPT entries. Numbers indicate the sequence of learned nodes.

Animal experiments show that a change of the natural order of developing senses disrupts e.g. in kittens the olfactory development [9] or in owls the auditory and visual spatial localization [4]. Also results from developmental psychology [10] as well as theoretical studies [1] indicate that an immature sensory system should not be seen as a bug but as a feature. Another interesting question is how the sensory development relates to the establishment of a hierarchy of categories, especially with respect to the coarse-to-fine progression.

1) Sensory Development

In the beginning only very restricted visual information about objects is available. More and more measurement results are incrementally added at later stages of development. If we relate this developmental process to the considerations made above about different knowledge acquisition strategies, it becomes clear, that human infants

follow a strategy that lies between the object-by-object and the property-by-property acquisition sequence: During the examination of an object the infant's sensory apparatus makes several measurements with respect to size, shape, color etc., but it has to reexamine this specific object (or another instance of the same category) after an improvement of sensory mechanisms has taken place or after new measurement capabilities have been established. Examples for quantitative and qualitative changes include the addition of detailed shape information due to an improved analysis of high spatial frequencies and the faculty of word learning which is not at all available at the very beginning.

2) Learning Direction

There has been a lot of dispute if human categories develop in a bottom-up or top-down fashion. Most researchers now agree that coarse categories like *object*, *food* and *indoor* are acquired before finer grained ones like *cup*, *banana* and *kitchen*. To relate this coarse-to-fine progression with the set of new nodes generated by our divorcing procedure, we have to take a closer look what these nodes represent and when they are learned. As illustrated in Fig. 4, the first node acquired via the learning schema is PROP1_PROP2. The automatically generated label of this node is arbitrary but it refers to what the node represents: objects that have properties PROP1 and PROP2. If we look at the sequence of acquired nodes (indicated by numbers in Fig. 4), we observe the same pattern described above, since an addition of properties leads to a more specialized representation.

B. Simulating Conceptual Development

We applied the child-friendly divorcing procedure to a small BN with 20 nodes and 40 links in order to simulate aspects of conceptual development. Motivated by the aforementioned animal experiments with normal order of sensory experiences vs. disrupted order, we will compare the categories that emerge from a biological ordering with the ones that emerge from a reversed ordering.

1) Stimulus Set

As stimuli we selected a balanced set of 8 objects and 12 properties. Fig. 5 depicts the exact pairing between nodes representing objects (001 to 008) and nodes representing properties (self-starter etc.).

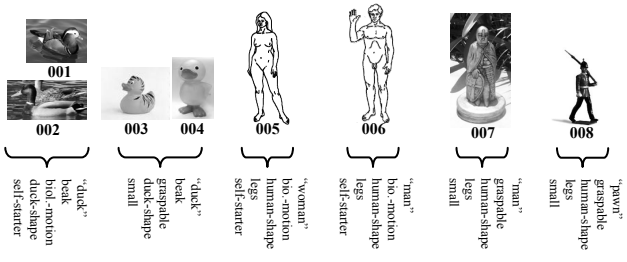

Fig. 5. Stimuli, their identifier and all their properties that were "measured" by the system. In total 40 object-property pairings were used. They get into the system as knowledge snippets like *006 has-property human-shape*.

2) Sensory Availability

To simulate the heterochronicity of measurement procedures the network was incrementally built in three separate phases. For empirical support of the different maturation times of sensory (sub-) modalities see e.g. [2].

In a first phase only the most important visual information is available: Motion can be detected and there is a functioning mechanism for the indication if the object in question belongs to the class of "agents". Furthermore, the system has very basic manipulative capabilities giving rise to crude size measurements. For the simulation here we used four exemplary properties in phase I:

- biological-motion
- self-starter
- small
- graspable

More advanced sensory measurements can be made in the next phase in which parts as well as shapes can be recognized. Thus we used the following properties in phase II:

- legs
- beak
- human-shape
- duck-shape

Finally, word for the set of learned objects are provided. For this we added labels in a third phase:

- "duck"
- "man"
- "woman"
- "pawn"

3) Resulting Categories

Table 2 shows all learned categories for the biological and reversed orderings (columns) and for each of the three phases (rows). To ease interpretation and avoid long node labels the categories are abbreviated with the following names: A node representing all objects of the domain that show a biological motion pattern and are observed to start by their own has the label BIOLOGICAL-MOTION_SELF-STARTER in the network and is abbreviated with *animal* in Table 2. Likewise a node with child nodes *graspable* and *small* can be said to represent a vague concept of small objects, therefore the node labeled GRASPABLE_SMALL in the network is in the table abbreviated with *utensil*.

Simulation results were obtained with 10 iterations in each condition. Numbers to the right of the category name therefore indicate the probability of occurrence. Sometimes nodes are generated which miss a property. A category label and the missing property with question mark indicate this.

It is interesting to see how the amount and variety of learned categories is influenced by the chosen sequence of available measurements. The biological ordering has led to a much richer representation as opposed to the reversed ordering, but more experiments are needed to elucidate this

effect. Instead, we focus here on representations that have or have not emerged: In the condition of a biological ordering, we see e.g. a representation that groups together any kind of *animal*, i.e. stimuli 001, 002, 005 and 006 (with probability of 1.0 in phase I, Table 2) but we do not see a representation that groups stimuli 001 and 002 with 003 and 004, which would be a joint representation for natural ducks and toy ducks. It is obvious that such a kind of representation is not desirable, since there is no common behavior that is applicable to these very different kinds of ducks.

	biological orderings (10 trials)		reversed orderings (10 trials)	
	learned category	p	learned category	p
PHASE I	Animal	1.0		
	Utensil	1.0		
PHASE II	Animal with beak and duck-shape =A1	0.7	Entity called "man" with legs and human-shape =E1	1.0
	Animal with legs and human-shape =A2	0.1	Entity called "duck" with beak and duck-shape =E2	1.0
	Utensil with beak and duck-shape =U1	0.4		
	Utensil with legs and human-shape =U2	0.4		
	A1 (but self-starter?)	0.2		
	A2 (but self-starter?)	0.3		
	U1 (but small?)	0.2		
	Utensil with beak	0.1		
	U2 (but small?)	0.6		
PHASE III	A1 called "duck"	0.4	E2 and Animal	1.0
	A1 (but self-starter?) called "duck"	0.1	E2 and Utensil	0.4
	U1 called "duck"	0.4	E2 (but label?) and Utensil	0.3
	U1 (but small?) called "duck"	0.1	E2 (but beak?) and Utensil	0.3

Table 2. Learned categories by applying child-friendly divorcing to the stimuli shown in Fig. 5. The outcomes of two settings that differ with respect to the availability of sensory measurements are shown for three hypothesized phases of network development. Numbers are probabilities that the respective categories were generated within ten trials. A1, U1 etc. are used as abbreviations.

This representation, which is avoided by the biological ordering, emerges if the incoming information is reversed: With a probability of 1.0 a node (*E2*) is learned that represents objects with the properties BEAK, DUCKLABEL and DUCKSHAPE. While there are also nodes learned for distinguishing between natural ducks (*E2 and animal*) and rubber ducks (*E2 and utensil*) in phase III, representations for the basic categories of *animal* and *utensil* are missing in the reversed orderings experiments.

V. Discussion

There are somewhat related approaches in fields like BNs, connectionism and neural networks. Here we can give only a short description of a few examples from these research areas.

There are many methods for learning probability models, but most deal with parameter learning. Algorithms that learn the net structure usually assume that the data are fully observable. A much harder problem is how to learn the structure from incomplete data efficiently, i.e. in the presence of missing values or hidden variables. One interesting approach is the structural EM algorithm [2], which combines parameter optimization with structure search. To tackle with this challenging problem some constraints are introduced: The number of hidden variables is fixed and it is assumed that they are on top (i.e. parents) of all the other nodes. In the much simple context described herein such a constraint would prevent the development of concept hierarchy.

In cognitive science the "Parallel Distributed Processing" approach [7] has attracted a lot of attention. They show how a simple computational approach can lead to a progressive differentiation of conceptual knowledge by exploiting the coherent covariations of objects' properties. Their work covers a whole bunch of simulations, but to our knowledge they do not take into account the changing availability of measurement procedures due to sensory development which is crucial to our approach.

References

[1] R.M. French, M. Mermillod, P.C. Quinn, A. Chauvin, D. Mareschal, "The importance of starting blurry: Simulating improved basic-level category learning in infants due to weak visual acuity", Proceedings of the 24th Annual Conference of the Cognitive Science Society, 2002, pp. 322-327.

[2] N. Friedman, "The Bayesian Structural EM Algorithm", Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence , 1998, pp. 129–138.

[3] P. Kellman, M.E. Arterberry, "The cradle of knowledge – Development of perception in infancy", MIT Press, 1998.

[4] E. Knudsen, "Instructed learning in the auditory localization pathway of the barn owl", Nature. 417 (6886), 2003, pp. 322-328.

[5] K. G. Olesen, U. Kjærulff, F. Jensen, B. Falck, S. Andreassen and S. K. Andersen, "A munin network for the median nerve - a case study on loops", Applied Artificial Intelligence, 3, 1989, pp. 384–403.

[6] OMCS, http://commonsense.media.mit.edu

[7] T. T. Rogers and J. L. McClelland, "Semantic Cognition: A Parallel Distributed Processing Approach", Cambridge, MA: MIT Press, 2004.

[8] F. Röhrbein, J. Eggert and E. Körner, "Prototypical Relations for Cortex-Inspired Semantic Representations", Proceedings of the 8th International Conference on Cognitive Modeling, Psychology Press, Taylor & Francis Group, 2007, pp. 307–312.

[9] J. H. Rosenblatt, R. Turkewitz and T. C. Schneirla, "Development of home orientation in newborn kittens", Transactions of the New York Academy of Sciences, 31, 1969, pp. 231-250.

[10] G. Turkewitz and P.A. Kenny, „The role of developmental limitations of sensory input on sensory/peripheral organization", Journal of Developmental and Behavioral Pedriatrics, 6(5), 302-306.