Selection for Representation in Higher-Order Adaptation

Solvi Arnold · Reiji Suzuki · Takaya Arita

Abstract A theory of the evolution of mind cannot be complete without an explanation of how cognition became representational. Artificial approximations of cognitive evolution do not, in general, produce representational cognition. We take this as an indication that there is a gap in our understanding of what drives evolution towards representational solutions, and propose a theory to fill this gap. We suggest selection for learning and selection for second order learning as the causal factors driving the emergence of innate and acquired forms of representation, respectively. Cognition is commonly viewed as a "black box"—selection works on externally visible behaviour alone, with little regard for implementation structure. Yet even if implementation structure is not constrained by selection on behaviour, implementation structure does affect how easy or difficult it is to make specific modifications to the behaviour. Hence selection for learning can affect the implementation structure of behaviour. Similarly, the implementation structure of learning ability itself is not under direct selection, but selection for second order learning can affect the implementation structure of first order learning. We argue that these indirect selection effects guide evolution towards representational implementations, as structural alignment between implementation structure and environment structure guarantees that simple changes in the environment can be met with simple changes in implementation. We illustrate the theory with examples of computational investigations, and discuss how the theory may help put representational cognition within reach of purely connectionist AI.

Keywords Representation · Connectionism · Second order learning · Evolution of learning · Evolution of mind · Cognitive map

S. Arnold (✉) · R. Suzuki · T. Arita
Graduate School for Information Science, Nagoya University, Furo-cho, Chikusa-ku,
Nagoya 464-8601, Japan
e-mail: solvi@alife.cs.is.nagoya-u.ac.jp

Introduction

Representation is one of the fundaments of advanced cognition. A theory of the evolution of mind cannot be complete without an explanation of how cognition became representational. However, a clear explanation seems to be missing. The issue is at the core of a long-running debate in AI. Proponents of a strongly representational view of mind have long criticized connectionist AI for its inability to account for the representational qualities of mind (Fodor and Pylyshyn 1988; Fodor and McLaughlin 1990; McLaughlin 2009). The root of this inability lies in the fact that connectionists typically use an automated adaptation process (evolution or learning) to structure their AI systems (e.g. simulated evolution of a neural network), and *somehow* our approximations of natural adaptation processes do not seem inclined toward representational solutions.

If we believe that mind is representational, and that it is the product of adaptation processes, then why don't our approximations of adaptation processes produce representational solutions? While this issue has gained prominence in AI and philosophy of AI, its relevance is by no means restricted to these fields. If evolution as we understand and model it does not produce representational cognition, then we are failing to understand something quite crucial about the origins of our own minds.

Correspondence and Representation

Representation is a vague term used by many different authors to denote many different things. We will not theorize much on the specifics of representation. Our focus is on its most basic fundaments, and under what sort of evolutionary conditions those may emerge.

At its most basic, we take representation to involve *correspondence relations* between elements of a cognitive system and elements of the task or environment it operates in (Spencer 1885, see also Godfrey-Smith 1996, 2002). We use the term correspondence here for the extent to which we can point to individual parts of a cognitive system and say what they are *for*, or what they *stand for*, or even what they *mean*. If for example some part (say a nerve cell) of a system selectively responds to vertical lines and another part selectively responds to horizontal lines, then we can point at these parts and say "*this* part handles vertical lines and *that* part handles horizontal lines". Or, when a system's operation involves the manipulation of linguistic expressions, then we can (at least in principle) isolate parts (words) and identify the environmental elements they stand for (whether this is sufficient for those parts to have *meaning* is a different question, that we will not concern ourselves with here). In such cases we say there are *specific* correspondence relations between the cognitive system and its environment (*one-to-one* correspondence). On the other hand if each individual part of the system is involved in handling many elements of the environment, and each individual element of the environment is handled by many parts of the cognitive system, then we cannot easily say what a given element is for, or say for a given part of the task or

environment which part of the cognitive system handles it. A visual system may work without any specific parts specialising on vertical or horizontal line orientations. In this case we have *fuzzy* correspondence between cognitive system and environment (*many-to-many* correspondence). Hand-coded AI systems usually feature highly specific correspondence, and theories about the nature of representation usually assume highly specific correspondence. Even when theorists disagree about the format of representations (e.g. whether they are pictorial or linguistic), they still share the basic intuition that representations are informational structures that consist of parts specifically corresponding to parts of the external world. As such, correspondence is more fundamental than the issue of representational format. It is on this issue of correspondence that we focus our attention.

Connectionist systems violate our intuitions about representation at this fundamental level. They are usually created using approximations of the adaptation processes that produced natural cognition (learning or evolution), but despite handling a variety of impressively complex tasks, they usually show very fuzzy correspondence only. This poses a bit of a paradox. Maybe it means the intuition that correspondence is important to cognition is just wrong? Some have indeed defended such positions (e.g. Brooks 1991). We think that conclusion is premature. There are still many tasks that connectionist systems struggle with, that hand-coded classical AI solves with ease by employing explicit and specific correspondence (in particular tasks that involve syntactic or similarly rule-based manipulations). So then maybe the problem is not in our intuition but in connectionism, for failing to account for correspondence. It has indeed been argued that connectionism just does not have much to contribute to the study of representational cognition (Fodor and Pylyshyn 1988). We think this conclusion, too, is premature.

We argue that the dearth of specific correspondence in connectionist systems undermines neither the importance of correspondence, nor the connectionist paradigm. We explain the paradox as follows: The reason representational cognition emerges in natural species but not artificial species is that the natural world *selects* for representational abilities, while AI's repertoire of tasks does not. No amount of adaptation is going to produce representational solutions when representation does not benefit the solution. In this light the paradox is really no more paradoxical than the absence of eyes in a species evolved in the dark. This simple explanation meets one obvious objection: cognition is a *black box*. What goes on inside is invisible to the environment, and hence the environment *cannot* select for representational architecture. This objection is wrong. Below we show how environments *can* select for representation, both theoretically and computationally.

We said connectionist systems are usually produced using approximations of evolution *or* learning. To combine the two is generally viewed as double the hassle with nothing to gain. This is a mistake. It is through the interaction between evolution and learning processes that correspondence enters the picture.

We will distinguish two classes of correspondence, *innate* correspondence (IC) and *acquired* correspondence (AC). As the terminology suggests, IC covers correspondence relations that are present at birth (or more precisely, are direct

products of a more or less fixed developmental process). Examples could be innate representations of predator species and other dangers, of what does or does not constitute food, or, more abstractly, the fundamental properties of space that one needs in order to make sense of sense data and coordinate one's motions. ICs are features of the species (roughly the same in each intact specimen), and are products of evolution. To understand the emergence of IC we should ask how an evolutionary environment can exert *selection pressure for IC*.

AC, on the other hand, covers correspondence relations that are acquired via learning processes. For example, many species are capable of forming map-like representations (known as "cognitive maps") of their surroundings to help them navigate. Another example are the representations used to navigate a dynamic and individual-specific social environment. ACs are features of individual specimen, and are acquired within the specimen's lifetime. They are products of learning, while the mechanisms responsible for acquiring them are products of evolution (note the stacking of adaptation levels here). To understand the emergence of AC we should ask how an evolutionary environment can exert *selection pressure for acquisition mechanisms for AC*. As IC and AC differ in the way they are selected for, they receive separate treatment below.

While both IC and AC can be viewed as (bases of) representation, we reserve the term *mental representation* for AC (we will return to this point later on).

Innate Correspondence

This section discusses the evolution of IC. We start with the theory, illustrate it with a toy example, discuss a computational investigation, and take a brief look at how IC features in natural cognition.

Innate Correspondence: Theory

Here we give a theory of the evolution of IC.

We define a behaviour (B) as a mapping from stimuli (S) to responses (R):

$$B : S \rightarrow R$$

We define learning (B') as a mapping from stimulus-behaviour pairs to behaviours, i.e. "stimulus-caused updates of behaviour":

$$B' : (S, B) \rightarrow B \ \ i.e.$$
$$B' : (S, S \rightarrow R) \rightarrow (S \rightarrow R)$$

This definition of learning is broad, including things one would not usually want to call learning. A blow to the head is a stimulus, and may cause a lasting change in behaviour via processes that bear little resemblance to learning processes. For our present purpose, however, it is enough that all learning processes fall under the definition. Note that any form of memory that affects behaviour, even very short

term, is also included under this definition. This is intentional; we do indeed consider such effects a form of learning here.

We also define the environment (E) as a mapping, one from responses to stimuli. An agent acts, and this (potentially) affects the subsequent stimulus it receives:

$$E : R \rightarrow S$$

Note that an environment is much like an inverse behaviour (mapping responses to stimuli instead of stimuli to responses). And just as behaviour may change, so may the environment, either as a result of the agent's responses or spontaneously:

$$E' : (R, E) \rightarrow E$$

We have defined behaviour, learning and environment as mappings, but organisms and environments are physical objects, not mathematical objects. In order for these mappings to exist in the physical world they must have implementations. For each of the mappings defined above, we let its lowercase partner denote its implementation: b, b', e, e'. The relation between mappings and implementations is one-to-many: just as infinitely many programs may realize the same input–output relation, infinitely many implementations may realize the same stimulus–response mapping. Implementations e and e' should be understood as the actual physical reality of the environment. The physical environment also determines the fitness effects of responses, but this information is not in general communicated to the organism. In reality, e and e' are generally not clearly distinguishable, but this is of little importance to our project.

Implementations b and b' are products of evolution, and as we said (theoretically) infinitely many options are available. We can expect evolution to favour those implementations that are *execution-efficient*, with respect to whatever resource restrictions the environment presents. A fast implementation if time is scarce, an "eco" implementation if energy is scarce, etc. As long as the mapping remains intact, evolution will cut whatever implementational corners it can (maybe even trading some deterioration of the mapping for a performance boost). As the history of connectionism illustrates, these conditions do not lead evolution towards representational solutions much, and indeed there is no reason to think they should. The goal criteria we have (making the right mapping and making it efficient) give direction to evolution, but nothing suggests that these are pointing towards correspondence (given the overhead representational solutions carry, efficiency may well direct evolution *away* from them). The stimulus–response box remains pretty black.

Learning updates behaviour on a within-lifetime timescale, and evolution updates both behaviour and learning ability on an evolutionary timescale. Figure 1 gives a graphical representation of the processes involved.

If the environment is dynamic in a sufficiently organized and predictable way as to make learning possible, then there is selection pressure on evolution of B'. Different implementations of B (i.e. different b) call for different implementations of B' (i.e. different b'). For example in the highly unnatural case that b would take the form of a table defining an output for each possible input independently, then b'

Fig. 1 Selection for IC. *Gray box* environment. *Black box* cognition. *Black arrows* update processes. *D* environmental update processes. *L* learning processes. *White arrows* regular selection pressures. *Gray arrow* diverted selection pressures. Letters on selection pressure arrows indicate what is selected for: *EE* execution-efficiency. *UE* update-efficiency. *B* behaviour function. *b* implementation of behaviour function. *B′* learning function. *b′* implementation of learning function. *E* environment function. *e* implementation of environment function. *E′* environmental change function. *e′* implementation of environmental change function

would operate by rewriting entries of this table. So whether and how feasible evolution of B' is strongly depends on the architecture of b. If there is selection pressure on B', then mutations in b that are beneficial to B' are beneficial mutations (even if they have no effect whatsoever on B). As an extreme scenario, we could imagine B remaining stable while b evolves to facilitate B'. This possibility shows that there is a fundamental difference between selection for a specific mapping and selection for a specific implementation of that mapping. Evolution working on B alone does not care much about the precise architecture of b, but evolution working on B' does. This gives us a possible explanation for how there can be selection pressure towards a specific b, without violating the "black box" intuition that b is hidden from the environment: as illustrated in Fig. 1, selection for specific implementation structure arises *inside* the black box, as part of the selection for B' "splits off" into selection for B'-friendly b.

B' constrains the structure of b, but we haven't said anything yet about what sort of b is favoured by B'. We will claim that B' benefits most from b that are in some sense isomorphic with the environment. The basic idea is as follows: If the environment and (consequently) the optimal behaviour are static, then difference in the structure of their implementations poses no problem. But if the environment and (consequently) the optimal behaviour may change (due to E' and B', respectively), then the more the structure of b and e differ, the harder it is for B' to update B in sync with E'. The implementations (e) of environments that cognition evolves in are composed of many distinct features (food sources, temperatures, other agents, spatial layouts, etc.) that act and interact to give rise to E. Let's call a change in one single feature a *simple* change. Simple changes in e often lead to *complex* changes in E: multiple input–output pairs change. Consequently a complex update of B is required. If b contains a feature that corresponds to the changed feature of e, in a functionally similar position, then the required complex change in B can be realized by a simple change in b. This makes B' quite feasible. If no such corresponding feature exists, a complex implementation update is required. In this case no straightforward relation exists between the environmental change and the appropriate behaviour change, making life difficult or infeasible for B'. So the

organization that evolves in b to facilitate B' should in one form or another capture the variable features of the environment along with their functional roles therein.[1]

Note that we neither claim that B' is strictly impossible without correspondence relations between the features of e and b (the issue is practical, not mathematical, infeasibility of optimal-yet-isomorphism-free b), nor that correspondence cannot occur in absence of B'. What we claim is that selection pressure on B' translates into selection pressure on b to form correspondence relations with fitness-relevant features of e, and that this "selection pressure conversion" is an organizing factor in the evolution of cognition, directing it towards representational solutions.

Two points of caution apply: (1) Even with such selection pressure in place, it is not guaranteed that correspondence will evolve. Selection for execution-efficiency may be stronger and, like we said, may point away from representational solutions due to their overhead. For correspondence to emerge, the fitness gains from update-efficiency must outweigh the costs incurred on execution efficiency, so we should think of scenarios where species can afford to carry around some brainpower and have substantial benefit to gain from efficient learning. (2) Note that in general, not all of b receives the organizing influence of diagonal selection. Innate behaviour that is impervious to modification by learning should not be affected. However, the parts of b that *are* modifiable by learning seem quite central to advanced cognition, and the theory provides a candidate explanation of why these parts should be organized as they are.

Innate Correspondence: Examples

Figure 2 shows (rather arbitrary) images depicting an environment function (E) and (two instances of) the target behaviour function for this environment (B^x and B^y). If it seems odd to depict functions like images, think of it as follows: the environment sends the organism stimuli in the form of coordinate pairs, and the organism reacts with one of two responses, say 0 or 1. The environment checks against its image whether the gray value at the location indicated by the stimulus is closer to black or to white. If closer to black, the organism receives a reward for responding 0, and if closer to black for responding 1. This means that if we lay out the ideal responses for all stimuli in the form of an image, with 0 shown as black and 1 as white, we get the image given by B^x and B^y. Figure 2 also gives the implementation structure (e) of E, and two possible implementation structures (b^x and b^y) for B (cognitions x and y). Cognition x composes the target behaviour out of four square pieces, arranging them side to side. Cognition y composes the target behaviour out of three differently shaped pieces, overlaying them on one another.

---

[1] Programmers may recognize something in this: If one codes a program for handling a fixed set of input data on a specific device, one can go and gain performance by cutting corners: exploiting the peculiarities of the data and device (common practice in early console game development). Such code gains efficiency at the cost of flexibility. On the other hand, code that needs to run on many different devices and/or may later have to be modified or expanded (possibly by a different programmer) is usually written with a fair bit of performance overhead, trading efficiency for modifiability. When mutation and selection are doing the programming, they will naturally take the former approach, unless forced to take the latter approach by environmental demands for modifiability.
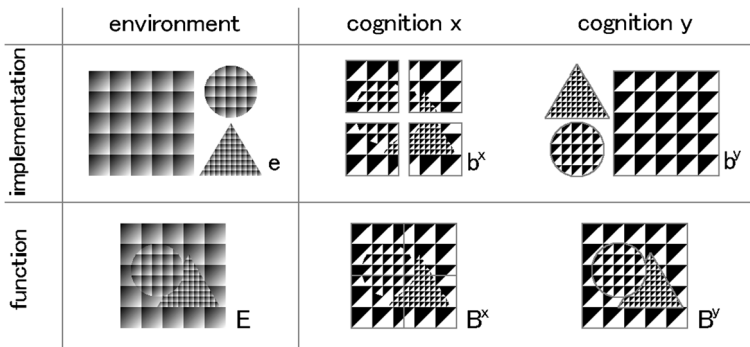
Fig. 2 Implementation and function of an environment and two behaviours. Under a static environment, correspondence-based implementation (y) has no advantage over other implementations (e.g. x)

If we ask whether x or y best realizes the target behaviour, the answer is that neither is better than the other; they implement the exact same function. In terms of execution efficiency, we may observe that if we add up the surfaces of all pieces, y has some overhead compared to x. The structure of cognition y reflects that of the environment, but there is no good reason why selection on the behaviour function alone should pick y from the infinite number of possible candidates.

Now let's say that this was just a snapshot, and that the environment is dynamic. Figure 3 is an extension of Fig. 2, now marked with time indices (i and j). Between i and j, the environment has changed. The figure also adds another cognition (z).

We see that the component parts comprising the environment have shifted around a bit, leading the environment function to change, from E_i to E_j. Now we put ourselves in the shoes of x and y's learning ability and try to adapt B^x and B^y to this change. (To be accurate, we would have to do this on basis of reward information we receive response by response, without directly observing E_j). It shall be obvious that updating cognition y will be fairly straightforward. In the case of cognition x however, we cannot make the update while retaining the given implementation structure. We would need to cut the pieces into pieces, and even so it would prove quite a challenge (note the change in overlap between the shapes). If we were learning ability, we would definitely have an easier time working with cognition y.

What gives the structure of cognition y the advantage is not that it is particularly suitable for implementing B_i; x is just as good for that (if not better). It is also not that it is particularly suitable for implementing B_j; z is just as good for that (if not better). The advantage of y is in the fact that it can match with minimal effort the way an environment with this structure changes. Hence its advantage only exists in the presence of learning ability.

Of course, there is one implementation structure that can match any environment without ever needing to cut up a single piece. If we simply bring a big pile of loose pixels (call this cognition p for short), we can match any target without concerning ourselves about structure. This is true, but consider what happens to the search space, and recall that we should be doing our updating on basis of a thin trickle of reward information. While a learning system working with cognition y is searching
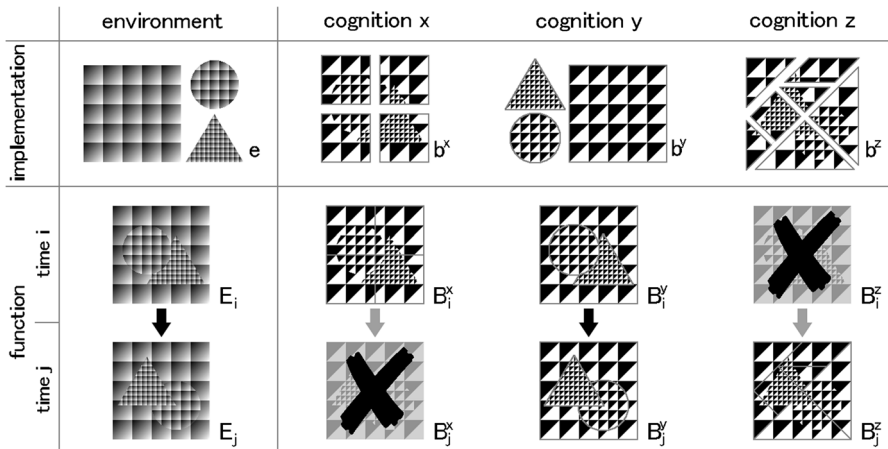
Fig. 3 Under a dynamic environment, correspondence-based implementation (y) has the advantage

the space of images composable of implementation y, a learning system working with the pixel pile would be searching the space of all possible images, a vastly larger space (notably, standard neural network learning algorithms would take the approach of cognition p). The situation can be likened to a game of Battleship,[2] where player y knows about the number of ships and their sizes, players x and z have mistaken beliefs about the ships and their sizes, while player p has no prior knowledge of any sort whatsoever.

Where, if anywhere, could we recognize IC in nature? Where IC results from selection for learning, one would expect it to be expressed in the form of "learning bias". Learning bias is the well-known phenomenon that when a test subject is presented with different learning tasks that have the same form and difficulty in abstract but differ in the actual stimuli and responses involved, performance will often differ substantially between those tasks, with higher performance on tasks that fit better with the test subject's natural context. Consider this example given by Gould and Gould (1994): it is easy to teach a rat to press a lever to receive a food reward, and easy to teach a rat to jump to avoid electric shock, but nearly impossible to teach a rat to jump for a food reward or press a lever to avoid electric shock, despite the fact that these tasks are identical in abstract. These observations make sense when we consider the rat's natural context: rats often procure food via manual manipulations, but never by jumping (rat food does not fly). The rat's cognitive architecture is such that the formation of new connections between manual manipulations and food reward occurs with relative ease, suggesting that in some sense, their learning ability relies on (tacit) knowledge about food. Or for another example, for a rat to form associations between an action and a reward usually

---

[2] A classic pen and pencil game where players arrange a fleet of "battleships" (lines of various lengths) on a grid, without revealing the arrangement to their opponent, after which they take turns "firing shots" at cells in the opponent's grid, trying to sink the opponent's ships by hitting all the cells occupied by every ship. Players inform each other only about hits and sunk ships.

requires the time span between action and reward to be short (on the order of seconds), with little or no intervening distractions. However, when rats experience sickness they will readily associate this negative reward signal with the taste of foods or liquids ingested hours before the onset of the sickness (the rat will develop an aversion to that particular food), but not with other stimuli, even if other stimuli precede the onset of sickness more closely (see e.g. Garcia et al. 1966). The rat's learning ability expresses presuppositions that sickness can be caused by food or liquids but not by, say, bells or lights, and furthermore that a substantial amount of time elapses between ingestion of bad food and the onset of sickness. Somehow the rat's learning ability exploits facts about rat physiology. The rat has no declarative knowledge of these facts, but its cognitive architecture is structured such that it is readily modified in accordance with these facts. In this implicit, operational sense, the rat's cognitive architecture expresses valid presuppositions about the way the world works. This characteristic gives the architecture an advantage over alternatives exactly because it supports learning (the presuppositions in question would have no use to a non-learner). Therefore it can be hypothesized to have evolved *as a result of* selection for learning. Our theory is that this alignment of a cognition's learning characteristics with the way the world actually works is underpinned by an alignment of structure, i.e. correspondence.

Innate Correspondence: Experimental Work

We have tested the theory laid out above using a computational model in which agents controlled by neural networks are evolved (using a genetic algorithm, see e.g. Goldberg 1989) to learn a simple food collecting task. A detailed description of the model and experiments can be found in Arnold et al. (2013). We necessitate evolution of learning by randomizing the relation between network output and the agents' actions. This produces a situation in which "newborn" agents are necessarily incapable of fit behaviour, as they don't know what signal to send to their "bodies" in order to make a given movement. They can, however, observe the effects of the signals they send to their bodies (their own displacement within the environment). The challenge for evolution is thus to devise a learning function (B') that takes this information and applies it to overcome the randomization of the relation between neural output and action. The neural networks were composed of neurons that control behaviour (this set of neurons corresponds to b in the above) and neurons that control the modification of connection weights between the first set of neurons (corresponding to b' above). The latter set of neurons operated on a modified version of the neuromodulation technique of Soltoggio et al. (2007, 2008). For comparison, we also evolved populations of networks to perform the same target behaviour innately, without randomization of the relation between neural output and action (i.e. no need for learning) and without the b' structures (i.e. no possibility of learning). This let us compare b structures evolved with and without selection for learning ability. If our theory is correct, then we should expect to see a difference in evolved b structures between these populations, with the latter showing a more diffuse neural organization while the former should show a structure that is optimized for modification by learning, incorporating structural features of the task.

Such effects were indeed observed. First off, variability of structure of *b* was markedly lower across trails (repetitions of the evolutionary process) when learning was required, indicating that selection for learning constrained evolution to a smaller set of implementation options. The *b* that evolved under selection for learning were also found to be more focused and compact, involving only a small subset of the available neurons, while without learning, *b* was seen to spread out diffusely over the available neurons. This compactness of the learners' solutions was a consequence of the way their learning operated. Their neurons were seen to specialize on specific choices between actions, using a simple principle. One can split the set of "possible situations" into subsets on basis of what action is the most appropriate response to it. The behaviourally relevant neurons each picked out two such sets of situations, responding negatively to situation belonging to one set and positively to situations belonging to the other. For example, we found a neuron activating positively when a step forward would be a good move and negatively when a left turn would be a good move. The distinguishing of situations was no finer than the crude action repertoire, but the neurons were picking out the features that called for "their" specific pair of actions with high precision, while leaving the features relevant to other actions to other neurons. This setup allowed the agent peculiar learning feats. For example, thanks to the aforementioned forward/left neuron, an agent finding itself turning left in a situation where a step forward would have been a good move would not just decrease its tendency to turn left in that or similar situations (thereby increasing its tendency to step forward instead), but it would also *increase* its tendency to turn left in (non-current) situations where a left turn actually is a good move, simply by assigning negative weight to the connection between this neuron and the active output neuron. Minimization of the size of *b* meant that every connection weight modification within *b* came to have such "side-effects" on other situations and actions. In an ideal *b*, these side-effects are advantageous (as is the one described), and this then becomes the target for evolution under selection for learning. The result is that the nets do not just stumble about randomly trying actions and learning only when something happened to work out well (as reinforcement-based learning would have them); they apply the information they gain via experimentation across situations they aren't currently in, learning as if they recognized their own action *as this or that action*. While the simplicity of the model does not allow for much display of intelligence, the nets certainly went about their task a little smarter than mere trial and error learning would ever have let them.

Lastly, and maybe most strikingly, *b* evolved under selection for learning showed a conspicuous symmetry, with the left and right halves of the visual input being handled with mirror versions of the same general connection patterns. Such symmetry was altogether lacking in *b* evolved in absence of selection for learning. That such a fundamental feature as symmetry only found its way into the cognitive architecture under selection for learning is maybe the clearest indication that selection for learning is a factor to take into account when discussing the evolution of cognitive architecture.

The theory so far provides a partial explanation for the origin of correspondence, and for why it is often lacking in neural nets. Evolution *can* make IC, but the

environment does have to select for it, and the environment does this when it selects for learning. The intuition that cognition is a black box to the environment is true at any given moment (any given stimulus–response exchange), but not over extended time when within-lifetime adaptation is involved.

It's worth noting that the interaction hypothesized here between learning and evolution can also occur between other pairs of adaptation processes, as long as we have one adaptation process at a larger timescale shaping another adaptation process at a shorter timescale. Crombach and Hogeweg (2008) show how lineage level evolution (large timescale) can speed up evolution at a smaller timescale in an environment with a cyclic dynamic (an effect known as "evolution of evolvability"). They observe that this acceleration is achieved by evolving what they call an "evolutionary sensor": a genome structure emerges that can track the cyclic environmental dynamic very efficiently, with mutation of just one or very few genome locations. Such "evolution bias" is straightforwardly analogous to learning bias, and evolutionary sensors could be said to *correspond* to the environmental state variables they track.

So far we have focused on IC, but how about AC? In the next sections we shift our focus (by one adaptation level), and discuss the evolution of mechanisms for within-lifetime acquisition of novel correspondences between a cognitive system and its environment.

Acquired Correspondence

In this section we extend the theory on IC to cover AC. We show how the theory applies to a well-known example of mental representation, and how the proposed theory lets us bring it within reach of connectionist modelling. We then briefly discuss our computational investigations.

From Innate Correspondence to Acquired Correspondence: Theory

We said that we view mental representation (MR) here as correspondence acquired via learning, that is, as correspondence produced by b' on a within-lifetime timescale from interaction between organism and environment. But as the perceptive reader will have noticed, b' falls outside the scope of the organizing influence of learning ability (as b' is itself not modified by b'). So the theory so far cannot capture MR. All we have so far is correspondence produced over the course of generations, by evolution.

But let's put that differently: we *do* have a mechanism for acquisition of correspondence, it's just that for explaining MR it operates on the wrong level: IC is *acquired* on the species-level, over the course of evolution. We want to make this process, of correspondence-acquisition, occur within each individual specimen instead. Luckily, for emergence of correspondence to occur, nothing much depends on the precise nature of the adaptation processes involved. We can just replace the evolution process (population-level adaptation) with a learning process (individual-level adaptation), and the learning process with, well, something else. Something

that relates to learning as learning relates to evolution: if correspondence-based b emerges under selection for B', then correspondence-based b' should result from selection for B'':

$$B'' : (S, B') \to B' \ i.e. :$$
$$B'' : (S, (S, (S \to R)) \to (S \to R)) \to ((S, (S \to R)) \to (S \to R))$$

So what is B''? B'' is a cognitive function that updates learning ability (updates the function that updates behaviour), in other words, B'' is second order learning (or "meta-learning"). Selection for *modifiability of learning ability* should result in correspondence-based learning ability, learning ability that operates by acquiring, on the fly, correspondence relations with the environment, or in other words, representations.

   We are not the first to suggest a deep link between second order learning and representational cognition. Approaching the issue from a different angle, Harry Harlow (in his seminal paper on learning sets) already wrote: "[L]earning to learn transforms the organism from a creature that adapts to a changing environment by trial and error to one that adapts by seeming hypothesis and insight" (Harlow 1949).

   Note that the interaction pattern suggested here involves three adaptation processes: evolution, learning, and second order learning. The evolution of second order learning guides the evolution of learning towards representational solutions.

   This somewhat cryptic proposition is shown in abstract in Fig. 4, but let us illustrate it with a concrete example. We will look at a well-known experiment designed to test for the ability to form and use "cognitive maps" (mental representations of spatial layouts). That is, the experiment is designed to expose the presence of AC to an outside observer, by teasing out action choices that cannot reasonably be explained without ascribing representational abilities to the test subject. "Outside observer" here would be the experimenter, but as action choice affects fitness, naturally occurring variants of the experiment expose representation ability to natural selection. Upfront, it might already be objected that such an experiment cannot exist. Strictly speaking, we indeed cannot infer representational structure from any sequence of actions. Yet intuitively, it is hard to imagine how the
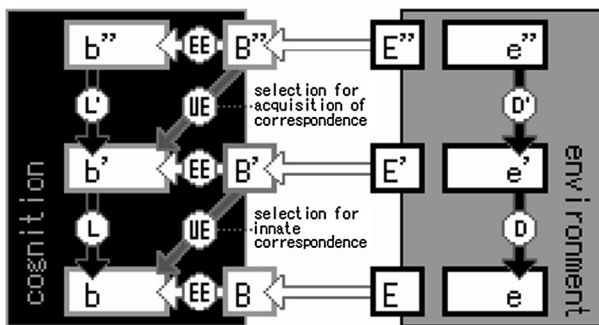

Fig. 4 Selection for correspondence acquisition (see caption Fig. 1 for details)

experimental task could be solved without a map-like representation. We show that the theory outlined above can reconcile fact and intuition: The action sequence that solves the task is proof not of representation, but of second order learning. Second order learning is theoretically possible but practically infeasible without correspondence, hence our intuition.

Tolman's Detour Maze

In experimental psychology, MR ability in biological species has often been studied using Tolman's detour mazes (Tolman and Honzik 1930). These mazes have multiple paths (at least three) from their start to their goal, varying in length (see Fig. 5). The shorter two paths join at some distance before the goal position. The experiment runs as follows: a rat is fed to satiation, then placed at the start of the maze. A food reward is placed at the goal position. The rat explores the maze, and eventually finds the food reward, but, being satiated, does not eat it. After the rat has thoroughly explored the maze, it is taken out (of the maze). We call this the exploration phase. Later, once the rat is hungry, it is placed again at the start position in the maze. The rat will now typically try to run the shortest path to the goal position and eat the reward. We call this the exploitation phase. This shows latent learning ability, but can still be explained as highly adaptive learning bias (the rat may have acquired an action sequence or state-action mapping that takes it from start to goal, without representing the spatial structure of the maze). However, in this phase MR ability can be revealed by blocking the shortest path and observing the rat's reaction. If the shortest path is blocked such that the medium path is still open (in Fig. 5: blocked at a cell with only a white dot) then the rat would ideally choose the medium path. If the shortest path is blocked such that the medium path is blocked too (in Fig. 5: blocked at a cell with both a white and a grey dot), then the long path is the correct choice. If the rat, upon encountering the blockage, backtracks to the start position and then (consistently over multiple trials) picks the new optimal path, then this taken as evidence of MR ability: If the rat had merely learned to solve the maze using an action sequence or state-action mapping, then finding one path blocked would tell it nothing about the viability of the other paths. So if it can pick the correct path right away, then it seems that it must have grasped
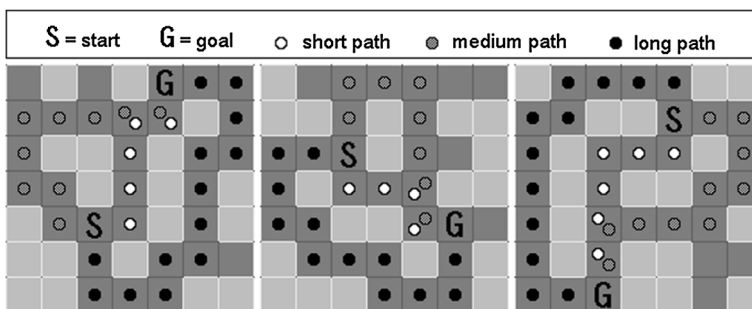

Fig. 5 (Randomly generated) detour mazes

the spatial relations between the paths. That is, it seems that it must have a spatial representation of the maze. Note that we recognize MR here by the *absence* of trial-and-error: we would not ascribe MR ability to the rat if it would need to try the other two paths to figure out which choice is now optimal.

Second Order Learning in Tolman's Detour Maze

Now we place the detour maze task in the theoretical framework introduced above. The maze task is composed of paths (or "accessible space", to be more precise), walls ("inaccessible space"), and a food reward. These are the aspects of e, implementing E. We see that a simple change in e (replacing one piece of accessible space with inaccessible space) leads to complex changes in E and consequently calls for a complex update of B (running a different path altogether). We also find ourselves strongly inclined to ascribe the ability to mentally represent spatial layouts to an animal if it can make this complex update of B *in an instant* (without further exploration) upon observing the blockage. We know that when we ourselves update our behaviour in such manner, we do so using our mental representation abilities. We said that mental representation is a form of AC. Our theoretical framework explains the evolution of correspondence-acquisition at b' as a consequence of evolution under selection for B''. Can we recognize B'' in the detour maze experiment?

When after blockage of the shortest path a rat infers the new optimal path without additional exploration, we can view this inference as a split-second B' process: a stimulus (observation of the location of the blockage) produced a change in behaviour (the rat abandons the blocked path and switches to the new optimal path). For B' to produce such a fast and effective and "context-sensitive" behaviour-update, B' itself must have been adapted to the maze (the update cannot be the result of fixed pre-existing learning ability, as the information in the observation alone does not suffice to explain the update without reference to the specific layout of this maze). The optimal update in behaviour has come to be causable by minimal information, and in this otherwise inexplicable efficiency of B', we observe the existence of a B'' process.

So what the outwardly observable behaviour of solving a detour maze shows us, strictly speaking, is not MR but second order learning. Why are we so inclined to ascribe MR when we observe B''? Because B'' calls for correspondence-based b', just like B' called for correspondence-based b in our discussion of IC. Maybe our intuition latches on to the feasibility argument more easily here, as now we are not talking about a function that updates a function, but about a function that updates a function that updates a function (the size of a tabular implementation would increase exponentially with update order).

We noted that the *absence of trial and error* (exploration) after the rat encounters the blockage is crucial for us to ascribe the rat MR. This link between MR and absence of trial and error has generality beyond the detour maze task, and necessarily runs via second order learning. If an organism performs all of its learning with fixed, innate learning ability, then the characteristics of the environment cannot actively be exploited to reduce trial and error. The innate

learning function of the *species* (B' unaltered by B'') may have been optimized to the evolutionary environment over the course of evolution, but in order for the learning ability of an *individual* to be optimized, on the fly, to the specifics of its current environment (e.g. a specific detour maze), there must be some within-lifetime adaptation process working on it. An adaptation process working on learning itself is by definition a second order learning process.

The view outlined above suggests a novel computational pathway into MR, which we discuss in the next section.

Second Order Neural Plasticity

We have computationally investigated second order learning in detour mazes, as well as in a simple social scenario. As in our computational work on IC, we forwent general-purpose first order learning algorithms, letting B' evolve instead. And with no general-purpose second order learning algorithms to forego, we left B'' up to evolution as well. We will outline our experiments shortly, but first let us discuss what is required for evolution of B''.

The neural basis of learning is neural plasticity. Given neural plasticity, learning ability can evolve, as happened in the computational work on IC. Would the neural basis of second order learning ability then be second order neural plasticity? What does that even mean?

In natural brains, the strengths of synapses ("connection weights") are modified not by an external update algorithm (as is common in AI), but they change as a function of the activation levels of the neurons they connect, modulated by various chemicals (neuromodulators). Release of neuromodulators is itself controlled by neural structures. Figure 6b gives a schematic representation of this concept. We may have a circuit that implements the current behaviour (b) and a circuit that controls the connection weight changes in that circuit (b'). This is first order neural
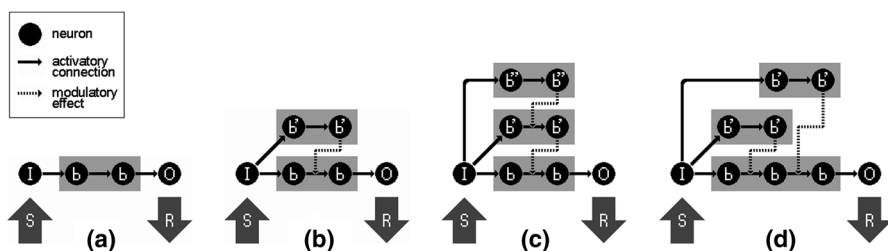

Fig. 6 Plasticity in neural circuits. Circuits receive stimuli (S) on their input neurons (I) and return responses (R) via their output neurons (O). A connection's plasticity is determined by the amount of modulation it receives. The images show (near minimal) examples of the schemas a neural network must contain in order for it to (potentially) express learning of a particular order using neuromodulation. **a** Circuit with fixed behaviour (no learning). **b** Circuit with first order learning. The b' circuit controls modification of the b circuit. **c** Circuit with second order learning. The b'' circuit controls modification of the b' circuit. **d** Alternative second order circuit. (Note that networks need not express these structures on a global level, and substructures within the schema can be elaborate networks in themselves. For the possibility of second order learning effects to exist we merely need a given schema to occur somewhere in the network. Furthermore, b, b' and b'' may overlap.)

plasticity. With it, we can make a system that modifies its own behaviour in response to stimuli (not just responding differently to different stimuli, but permanently modifying its stimulus–response mapping in response to stimuli). That is, B'.

We can get to second order neural plasticity simply by adding another circuit (b'') that modifies the connection weights in the b' circuit (Fig. 6c). Modifying the b' circuit effectively modifies the way the b circuit is modified, so using b'' circuits we can make systems that can modify the way they modify their behaviour in response to stimuli. That is, B''. The schema of Fig. 6d shows an alternative where plasticity loci are linked serially instead of in parallel. Though less intuitive, this too can produce second order learning effects.[3]

Neuromodulation-mediated connection weight change is not the only option for lasting behaviour modification. Connection "loops" that hold on to activation over time (as used in recurrent neural networks) can do the job too, and circuits with such loops too can be stacked to create second order update dynamics. In natural brains, neuromodulation-mediated connection weight change is associated with long-term memory and neural "echo" loops with short-term memory, but the distinction is not of importance here.

If our theoretical framework is correct, then availability of second order plasticity circuitry to an evolving neural system (i.e. existence in the search space of networks containing circuits with second order plasticity) is crucial to evolution of MR, and any successful solution must feature second order plasticity. As discussed above, MR makes B'' feasible, so selection for B'' points towards MR. But for that to happen, the species must be *susceptible* to selection for B'', and in order to be susceptible to selection for B'', it needs the *possibility* of second order neural plasticity circuitry—a possibility that is typically missing in AI systems.[4] Again we discuss some computational work, in which this time we looked at what happens when we make second order plasticity available to a population of neural networks evolving in environments that select for second order learning.

B'' and Correspondence: Experimental Work

We have tested our ideas in two pieces of computational work. This first targets spatial representation, the second social representation. Detailed descriptions of the experiments can be found in Arnold et al. (2012, 2014), respectively.

---

[3] To see how a circuit like this can produce second order change at the stimulus–response level, consider what happens with either b′ circuit's ability to modify the behaviour when the *other* b′ circuit sets the connection weight it controls to zero: the circuit loses its ability to modify behaviour. Hence modifications by either b′ circuit can affect whether or not the other circuit can modify behaviour. By the same token more subtle second order plasticity effects are possible too.

[4] Take for example the error back-propagation algorithms. This is a class of algorithms that adjusts connection weights so as to bring the output for a given input pattern closer to some target output for that input pattern. Algorithms of this class are also at the core of many implementations of Reinforcement Learning. In error back-propagation, the whole network acts as a single plasticity locus, all weight changes geared to modify behaviour, with no modification of the learning process itself.

Evolving Spatial Representation

We made a population of simple agents, living on tiny grid worlds, capable of moving to adjacent cells in the four cardinal directions, and of observing the accessibility of adjacent cells. We gave the agents randomly generated neural networks for brains, meaning they were initially incapable of any useful behaviour whatsoever, spending most of their lives bumping into walls helplessly. We presented them with a variety of simple navigation tasks, as well as the full-fledged detour maze task as described above. The simpler tasks were included to get evolution on its way, selecting for basic abilities such as walking a path or remembering a location.

It should be said that the neural networks serving as brains were no everyday garden variety NNs. We introduced some structural features based on the neurology of cognitive maps in rats (in particular, we provided uniformly connected grids of neurons as building blocks, inspired by the structure of the rat hippocampus), as well as various neurotransmitters. These made it comparatively easy for memory structures to emerge. The main feature for our purpose however was again the use of neuromodulation, and of echo loops, as these make possible the evolution of circuits with second order neural plasticity. We let the nets evolve using a simple genetic algorithm.

We found that provided with this possibility of second order learning, the networks evolved to solve the detour maze task. Their solutions prominently featured acquisition of correspondence (encoding the layout of the maze in either connection weight patterns or neural activation patterns), as one would intuitively expect from anything that solves a detour maze. The setup of the neural nets facilitated evolution of such solutions, but the fact that they did indeed evolve shows that the nets were being subjected to selection for mechanisms for correspondence acquisition. As we argued, correspondence acquisition is not something an environment can directly select for. What we did select for, with the detour maze environment, was second order learning. The possibility for second order plasticity made the nets susceptible to this selection pressure. Indeed, all observed solutions crucially relied on circuitry of at least second order plasticity for solving the detour maze task: they used second order learning to solve the task.[5] But second order learning is practically infeasible without correspondence acquisition. Hence direct

---

[5] We predicted that the detour maze task *can* be solved with second order learning, and *cannot* be solved with first order learning alone. Our computational results confirm the former and corroborate the latter prediction, but our experiments of course cannot prove that the task is strictly unsolvable without second order plasticity (the fact that we see no such solutions does not imply that they do not exist). Some readers may object that detour mazes have been solved using Reinforcement Learning (RL) algorithms, which we said are restricted to first order learning. As far as we are aware of, RL solutions to detour mazes and other latent learning tasks invariably extend the core RL algorithm with additional mechanisms to capture structural aspects of the task in some form or another (see e.g. Voicu and Schmajuk 2002; Gerard et al. 2005), and use information gathered by that mechanism to *guide the learning process*. In other words, such approaches hard-code a representational faculty, and use it to realize second order learning. The necessity of such mechanisms for solving detour maze tasks with RL corresponds directly to the need for second order plasticity in our model.

selection for second order learning ability put the nets under indirect selection for mechanisms for acquisition of correspondence, as the theory predicts.

Evolving Social Representation

Next we evolved a very simple form of social cognition. Agents are evolved to (1) perform a base behaviour and (2) predict the actions of a partner agent performing the base behaviour. For the base behaviour agents must pick actions in accordance with the current environment and their current mental state (a small bit string). Optimal behaviour here is set arbitrarily, agents simply have to match it. When predicting others' actions, agents see the environment but not the mental state of the partner agent. Hence the prediction task requires agents to observe the partner agent and learn (they have to "get to know" the other agent before they can predict its behaviour in various environments). Networks are evolved to solve this task, using again a simple neuromodulation mechanism to allow for evolution of the requisite learning ability. This time we avoided any pre-imposed structure.

    We let the mental state of the partner change frequently, but only one bit at a time. This produces a second order learning task (unsolvable with supervised learning). Analysis of the networks evolved under these conditions show that they learn by decoding the partner's mental state from its behaviour. Small sets of connections in these nets are found to encode the individual bits of the partner's mental state, allowing us to read (and even modify) individual bits of knowledge. These solutions show highly specific AC.

    In a set of control experiments, we kept the mental state of the partner agent constant over the duration of the interaction. This produces a first order learning task (essentially supervised learning). Bit-level representation of the partner's mental state is not evident in nets evolved using this task. Instead these solutions show effective learning with fuzzy correspondence.

    These findings lend further support to the idea that evolutionary selection for second order learning is selection for cognitive mechanisms for acquiring specific correspondence.

Discussion

We have discussed two types of correspondence, innate and acquired, and how they may emerge from interacting adaptation processes. Early on we noted that we reserve the term *mental* representation for AC. Having discussed examples of IC and AC, it may now be clear why we choose this terminology. In the examples we considered, the correspondences that we theorize to underlie learning bias are likely opaque to the organism implementing them. The fact that, in a rat's natural environment, a manual manipulation is more likely than a jump to produce a food reward is expressed in the rat's learning ability, but we have little reason to ascribe the rat epistemic access to this fact. On the other hand, if we think of cognitive maps and other such ACs, then it's hard to see how they could serve their function without their host having epistemic access to them. The question whether this is a qualitative

difference between innate and acquired correspondence or merely an artefact of our choice of examples is left for another occasion (hence the terminology should be considered tentative).

The theory suggests a deep relation between second order learning and MR, but this relation is a subtle one, so let us stress the following: We are not saying that MR "is" second order learning (whatever that would mean). We are saying that selection for second order learning partially diverts into selection for correspondence acquisition abilities (for feasibility of second order learning). Such correspondence acquisition abilities are representation abilities (and the correspondences acquired are representations). This justifies the inclination to ascribe an organism MR abilities when we see it solve second order learning tasks (such as Tolman's detour mazes or Harlow's learning set tasks): ability to solve second order learning tasks indicates that the species has evolved under exposure to selection for correspondence acquisition abilities.

Although one may need second order tasks to elicit external indication of MR, we should not infer that MR is the evolutionary product exclusively of detour mazes and other scenarios that strictly require second order learning to be solvable. Any scenario that selects for speedy learning selects for second order learning, just like any scenario that selects for speedy locomotion selects for acceleration.[6] Hence selection for second order learning may be more ubiquitous than it would seem if we only consider explicitly second order tasks.

Let us now briefly return to the debate noted in the introduction, surrounding connectionism's viability as a paradigm for studying mind. Representation is a crucial feature of advanced cognition, but connectionist systems generally fail to show representationality. This isn't because connectionism is unfit as a paradigm for modelling mind. It's because connectionist systems generally aren't exposed to selection for representationality, while natural cognition is. We deem it a virtue of the paradigm that connectionist systems can work in both highly diffuse ways and highly representational ways, as this means we can use them to evaluate the effects of all sorts of environments and selection pressures on their representationality, and thereby learn about the evolutionary origins of mind. However this potential has remained largely untapped. It is instructive to ask why.

We think the core cause may be that, despite its cognitive ambitions, connectionist AI suffers from a certain implicit behaviourism in its attitude towards learning (see also Balkenius 1994). Many research efforts are focused on developing *generally applicable* learning algorithms. This sounds great in theory. Surely algorithms that can solve many problems are "better" than algorithms that can solve only one. Error back-propagation and reinforcement learning algorithms are the big deal they are exactly because they can solve vast ranges of problems. However, generality also implies that the specific contingencies of a particular problem cannot be exploited in solving it. This means that a general algorithm is necessarily suboptimal with respect to almost all problems, and that it will fail to capture what nifty

---

[6] We can of course extend this observation to higher orders of learning. Whenever there is selection on speedy second order learning, there is selection on third order learning, too. How important orders above the second are is an open question (though we suspect that increasingly higher orders rapidly drop off in evolutionary impact and explanatory importance).

cognitive machinery may do the exploiting (learning bias and mental representation included). It might seem that one cannot have it both ways. However, at least in principle, one actually can avoid both the need to hand-craft specific algorithms and the inherent sub-optimality of general algorithms. Namely, by letting a general algorithm do one of the few things you actually need generality for: making specific algorithms. This is exactly what evolution does when it makes learning abilities, but somehow this fact remains sorely underlit in the field (for notable exceptions see Chalmers et al. (1990), Nolfi et al.(1994), Nolfi and Parisi (1996), Oiko et al. (2005), and the neuromodulation work of Soltoggio et al. (2007, 2008) that our own models borrow from). Instead, evolution and learning are usually heaped together as different ways of doing the same thing (adaptation) and combining the two is viewed as double the hassle with little to gain. If connectionism is to shed light on learning (and, if we are correct, representation) as it occurs in nature, then it should view learning ability as not just subject but also object of adaptation.[7]

Conclusions

We discussed how correspondence between a cognitive system and its environment may evolve as a result of selection for learning ability, and how mechanisms for acquisition of correspondence may evolve as a result of selection for second order learning. In both cases, correspondence is the result of the same interaction pattern between adaptation processes (an interaction pattern also seen in evolution of evolvability). Exactly how far these effects go in explaining the representational features of advanced cognition in nature is an open question. Maybe these are just two among many correspondence-producing effects at work in nature, and maybe they are minor in comparison to others. But if anything, we hope to have shown that it is a cogent and productive to ask how environments select for representational cognition, and that in doing so we can make the evolution of representation amenable to computational study.

---

[7] Traces of such a view of learning have emerged in the field of Artificial Life. Here interaction between evolution and learning has become a topic of much research since Hinton and Nowlan (1987) first demonstrated the Baldwin effect in a computational model. The focus is almost invariably on the effect learning has on evolution, but specialized forms of learning can easily emerge during the genetic assimilation phase of the Baldwin effect. If the phenotype has multiple loci that can be modified by learning, and these loci have their degrees of plasticity defined independently in the genotype, then genetic assimilation proceeds by lowering the plasticity of individual loci as plasticity on those loci becomes unnecessary. This leads to adaptively non-uniform plasticity distribution in the phenotype, which is expressed as advantageous learning bias. Especially if the optimal solution involves a limited but non-zero amount of plasticity, evolution of highly specialized forms of learning can be observed (see e.g. Suzuki and Arita 2004; Arnold et al. 2010).

References

Arnold, S. F., Suzuki, R., & Arita, T. (2010). Evolving learning ability in cyclically dynamic environments: The structuring force of environmental heterogeneity (abstract). *Proceedings of the twelfth international conference on the simulation and synthesis of living systems* (pp. 435–436). Cambridge, MA: MIT press.

Arnold, S. F., Suzuki, R., & Arita, T. (2012). Second order learning and the evolution of mental representation. *Proceedings of the thirteenth international conference on the simulation and synthesis of living systems* (pp. 301–308). Cambridge, MA: MIT press.

Arnold, S. F., Suzuki, R., & Arita, T. (2013). Selection for reinforcement-free learning ability as an organizing factor in the evolution of cognition. *Advances in Artificial Intelligence, 2013*,13. doi:10.1155/2013/841646.

Arnold, S. F., Suzuki, R., & Arita, T. (2014). Using second order learning to evolve social representation (Theory of mind). *Proceedings of the 8th international conference on bio-inspired information and communications technologies*.

Balkenius, C. (1994). *Biological learning and artificial intelligence*. Paris: Lund University Cognitive Studies.

Brooks, R. A. (1991). Intelligence without representation. *Artificial Intelligence, 47*(139–160), 1991.

Chalmers, D. J., Touretzky, D. S., Elman, J. L., & Hinton, G. E. (1990). The evolution of learning: An experiment in genetic connectionism. *Proceedings of the 1990 Connectionist Models Summer School* (pp. 81–90). Morgan Kaufmann.

Crombach, A., & Hogeweg, P. (2008). Evolution of evolvability in gene regulatory networks. *PLoS Computational Biology, 4*(7), e1000112. doi:10.1371/journal.pcbi.1000112.

Fodor, J., & McLaughlin, B. (1990). Connectionism and the problem of systematicity: Why Smolensky's solution doesn't work. *Cognition, 35*, 183–204.

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

Garcia, J., Ervin, F. R., & Koelling, R. A. (1966). Learning with prolonged delay of reinforcement. *Psychonomic Science, 5*(3), 121–122.

Gerard, P., Meyer, J. A., & Sigaud, O. (2005). Combining latent learning with dynamic programming in the modular anticipatory classifier system. *European Journal of Operational Research, 160*(3), 614–637.

Godfrey-Smith, P. (1996). *Complexity and the function of mind in nature*. Cambridge: Cambridge University Press.

Godfrey-Smith, P. (2002). Environmental complexity and the evolution of cognition. In R. Sternberg & J. Kaufman (Eds.), *The evolution of intelligence* (pp. 233–249). Mahwah: Lawrence Erlbaum.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization and machine learning*. Boston, MA: Addison-Wesley Longman.

Gould, J. L., & Grant Gould, C. (1994). *The animal mind*. New York: Scientific American Library.

Harlow, H. F. (1949). The formation of learning sets. *Psychological Review, 56*, 51–65.

Hinton, G. E., & Nowlan, S. J. (1987). How learning can guide evolution. *Complex Systems, 1*, 495–502.

McLaughlin, B. P. (2009). Systematicity redux. *Synthese, 170*, 251–274.

Nolfi, S., & Parisi, D. (1996). Learning to adapt to changing environments in evolving neural networks. *Adaptive Behavior, 5*(1), 75–97. doi:10.1177/105971239600500104.

Nolfi, S., Parisi, D., & Elman, J. L. (1994). Learning and evolution in neural networks. *Adaptive Behavior, 3*(1), 5–28. doi:10.1177/105971239400300102.

Oiko, T., Suzuki, R., & Arita, T. (2005). Evolution of dynamic role differentiation based on phenotypic plasticity in multi agent system. *Proceedings of the computational science symposium* (pp. 201–208).

Soltoggio, A., Bullinaria, J. A., Mattiussi, C., Durr, P., & Floreano, D. (2008). Evolutionary advantages of neuromodulated plasticity in dynamic, reward-based Scenarios. *Proceedings of the eleventh international conference on the simulation and synthesis of living systems* (pp. 569–576). Cambridge, MA: MIT Press.

Soltoggio, A., Durr, P., Mattiussi, C., & Floreano, D. (2007). Evolving neuromodulatory topologies for reinforcement learning-like problems. *Proceedings of the IEEE congress on evolutionary computation* (pp. 2471–2478). doi:10.1109/CEC.2007.4424781.

Spencer, H. (1885). *The principles of psychology* (3rd ed.). New York: Appleton.

Suzuki, R., & Arita, T. (2004). Interactions between learning and evolution: The outstanding strategy generated by the Baldwin effect. *Biosystems, 77*, 57–71.

Tolman, E. C., & Honzik, C. H. (1930). "Insight" in rats. *University of California Publications in Psychology, 4*, 215–232.

Voicu, H., & Schmajuk, N. (2002). Latent learning, shortcuts and detours: A computational model. *Behavioural Processes, 59*, 67–86. doi:10.1016/S0376-6357(02)00060-8.