

Seed-Programmed Autonomous General Learning

Kristinn R. Thórisson

THORISSON@RU.IS

Department of Computer Science, Reykjavik University & Icelandic Institute for Intelligent Machines Reykjavik, Iceland

Editor: Minsky, H. and Robertson, P. and Georgeon, O. L. and Minsky, M. and Shaoul, C.

Abstract

The knowledge that a natural learner creates based on its experience of any new situation is likely to be both *partial* and *incorrect*. To improve such knowledge with increased experience, cognitive processes must bring already-acquired knowledge towards making sense of new situations and update it with new evidence, cumulatively. For the initial creation of knowledge, and its subsequent usage, expansion, modification, unification, compaction and deletion, cognitive mechanisms must be capable of self-supervised “surgical” operation on existing knowledge, involving among other things self-inspection or *reflection*, to make possible selective discrimination, comparison, and manipulation of *newly demarcated subsets* of any *relevant part* of the *whole knowledge set*. Few proposals exist for how to achieve this in a single learner. Here we present a theory of how systems with these properties may work, and how cumulative self-supervised learning mechanisms might reach greater levels of autonomy than seen to date. Our theory rests on the hypotheses that learning must be (a) organized around causal relations, (b) bootstrapped from observed correlations and analogy, using (c) fine-grain relational models, manipulated by (d) micro-ampliative reasoning processes. We further hypothesize that a machine properly constructed in this way will be capable of *seed-programmed autonomous generality*: The ability to apply learning to any phenomenon – that is, being domain-independent – provided that the seed reference observable variables from the outset (at “birth”), and that new phenomena and existing knowledge overlap on one or more observables or inferred features. The theory is based on implemented systems that have produced notable results in the direction of increased general machine intelligence.

Keywords: Cumulative Learning, Self-Supervised Learning, Seed Programming, Machine Learning, Knowledge Representation, Autonomous Generality

1. Introduction

Many examples of self-supervised autonomous learning are found in nature—in fact, a notable part of adaptive behaviors in individual organism with a brain is acquired through experience, to various extents, over varying periods of time. Of special interest to artificial intelligence (AI) researchers are methods that would allow a machine to efficiently and effectively amass such adaptations, as well as generalize these over time, to be applied as appropriate in future circumstances.

Common to all cognitive learners in nature is that they started from a seed and grew into thinking beings. Out of the path that brings a single cell to a fully grown animal, we address here only the subset of how a newborn learner may bootstrap its knowledge in the

environment it is born. How can a learner autonomously generate knowledge of itself and its environment starting from an initial and necessarily small “knowledge nugget”—a kind of cognitive seed? How does it hold on to and use what it learns over time, in a way that future learning can be built on? And how does it modify it, in light of new information? Such *autonomous cumulative learning* is markedly unlike most machine learning (ML) methods developed to date and is not well understood at present (Thórisson et al. 2019).

This paper presents a theory of how self-supervised cumulative learning of novel phenomena may be realized. We consider the conjecture that knowledge bootstrapping at birth is a special case of the general principles involved in bootstrapping learning in partially-unknown, *novel*, circumstances. In both cases a learner starts with something given that is insufficient for addressing the novelty, facing the cognitive task of making use of what it already has to make sense of the new (Wang 2013). To address this subject we must look at three interlocked co-dependent realms: (a) The world of the learning agent and its target *task-environments*; (b) the mechanisms for control and management of the *cumulative learning* process; and (c) how learning is bootstrapped through an existing program—a *knowledge seed*.

The mechanisms of learning found in nature¹ are the result of requirements imposed by a non-axiomatic and combinatorial nature of the physical world itself. Its underlying rules can only be inferred from observations of the transformations of measurable variables accessed through sensory organs. Yet such learning happens in a self-supervised manner—without the help from a parent, teacher, paper-and-pencil, pocket calculators, or other outside assistance. Individuals of many animal species display robust autonomous properties that could benefit artificial learners, including learning many tasks over time, transferring knowledge between situations (transfer learning), handling distractions, learning many tasks (multi-goal learning), and handling novelty. Yet most of these remain nonexistent in modern machines. The title of this paper could well have been “machines that figure out how to get new stuff done” because an intelligence is useless if it cannot do something useful; to get things done requires knowledge of cause and effect – what affects what and how – and if there is no “new stuff” to be dealt with intelligence isn’t really called for, as once it’s figured out it could be stored and applied from memory. An intelligent agent will of course access its memory to perform tasks (humans do), but how necessary is intelligence for that?

At its simplest, learning a single well-defined task incrementally might be accomplished in a machine using methods and steps provided a priori by a human coder. Should some part of a task or environment be unknown beforehand, however, autonomous methods for incremental knowledge creation and unification would be needed. An important part of cumulative learning is relating relevant existing knowledge to new information, as for instance when the features of a particular outdoor sport, say tennis, are related to a different sport, e.g. football, with which it shares some similarities, allowing a quicker and more coherent knowledge creation and subsequent understanding of the new phenomenon. A large part of such processes must involve analogies (cf. Sheikhlar et al. 2020, Besold and Schmid 2016) – not necessarily the kinds that we make consciously when reading poetry or

1. To keep the discussion focused, we consider human cognitive abilities a key example of what we would our artificial agents to reach, but some household and even wild animals have demonstrated abilities worthy of consideration (cf. Balakhonov and Rose 2017, Patterson and Gordon 2001) and present in some ways a more appropriate comparison at the present stage of AI development.

playing compare-and-contrast games, but rather more automatic – such that the relevant features and similarities to be identified and analyzed are automatically highlighted, selected and processed. As we shall argue, in addition to analogies the process also involves three other forms of reasoning, deduction, abduction and induction.²

The more *general* a cumulative learner is, the more diverse tasks and environments it should be capable of learning, other things being equal, and the more diverse its acquired knowledge can become over time. Such learning, if appropriately implemented (a focus of this paper), should make an agent increasingly capable of handling a growing number of unfamiliar situations—without blind experimentation. For that to be possible, for any and every situation the learner may find itself in, a sufficiently effective and efficient process must continually assess the relevance of already-acquired knowledge and ensure its strategic application.

While some of the topics mentioned, e.g. generality and autonomy, are already in the cross-hairs of some research programs (cf. Georgeon and Riegler 2019, Reid et al. 2018, Lawless et al. 2017, Besold and Schmid 2016), relatively few aim explicitly to create a single learner that captures them all. Yet that is what is required to realize general machine intelligence. One reason for lack of progress on this point may be the perceived difficulty of addressing all at once the large set of requirements that such systems would need to meet (cf. Thórisson et al. 2015, Laird and Wray III 2010). Applying a more or less reductionist approach may seem at first glance to be in accordance with the scientific method, and it may suffice to study some parts in isolation, but to uncover the workings of complete complex systems this approach is insufficient: To take an example, as anyone familiar with the internal combustion engine will recognize, in systems that rely on a large set of interlocked, intricate mechanisms that produce a coordinated output, the mechanisms must be tuned to each other for the whole system to work—even a tiny deviation in one part may cause a breakdown in another. This principle explains in part why medical diagnosis is difficult; the principle is equally important when reverse-engineering other phenomena such as thinking. Dissecting the mind into small pieces and studying them separately may produce some results, but these will not suffice for understanding of how the system works as a whole, forever prevented by incongruent isolated theories, each resting on background assumptions orthogonal to others in one or more ways—any hope of ever piecing together a pot-pourri of parts developed this way into a coherent architecture is wishful thinking (cf. Thórisson 2012, Thórisson 2008, Garlan et al. 1995). What is called for is a panoramic perspective resting on a coherent global methodology that avoids applying the scientific method in a way that – in a call for complexity reduction – results in the exclusion or elimination of one or more elements that are integral to the nature of the phenomenon under study.

Answering Newell’s (1994) call for unified theories of cognition, our theory outlines how all these properties may be unified in an effort to realize artificial agents with general intelligence. The theory follows a *constructivist-AI methodology* approach (CAIM) outlined in Thórisson (2017, 2012) and combines several threads in my and collaborator’s work over the past ten years on the topics of *cumulative learning* (Thórisson et al. 2019, Thórisson

2. It may be useful to think of these as low-level (subconscious) processes—we use the prefix “micro-” because in our theory they are fine-grain and fast and much of their operation may not be readily accessible through familiar conscious introspection methods.

and Talbot 2018a, Thórisson and Talbot 2018b), *autonomous transfer learning* (Sheikhlar et al. 2020), *self-programming* and *autonomy* (Nivel et al. 2014b, Nivel et al. 2014a, Nivel and Thórisson 2013a, Nivel and Thórisson 2008), *bounded recursive self-improvement* (Nivel et al. 2013b), *understanding* (Bieger et al. 2017, Thórisson et al. 2016b, Steunebrink et al. 2016, Nivel et al. 2013b), and *task theory* (Eberding et al. 2020, Bieger and Thórisson 2018, Thórisson et al. 2016a). While the progression and handling of *time* is an essential and integral part in our theory (cf. Nivel et al. 2013b, Helgason and Thórisson 2012, Helgason et al. 2013), and goes hand-in-hand with the above topics, space does not allow us to go into any detail on that here.

The theory addresses a number of prior AI and philosophical questions, as well as important questions for autonomous general intelligence including continuously interruptable and self-correcting planning, flexible prediction generation and learning by anticipation, life-long continuous learning, flexible goal-driven behavior and goal-driven learning, cumulative learning, autonomous sub-goal runtime generation, multi-goal learning and operation, all of which will be addressed here. Other important related topics such as symbol grounding, explanation generation, the frame problem, and the temporal credit assignment problem, which the theory also covers, will have to wait.

Most of the concepts and principles we describe have been implemented and tested, demonstrating results that exceed other learners on important dimensions including sequence learning, multi-task learning, natural language interpretation and continuous life-long learning (cf. Nivel et al. 2014c, Nivel et al. 2013a, Hammer and Lofthouse 2020, Wang 2007), but especially with respect to realizing all of them in a unified parsimonious way in a single realtime learner. A closely related system that rests on a highly compatible theoretical basis is Wang’s non-axiomatic reasoning system (NARS; Wang 2006, Wang 2013). This, as well as relevant work of others, will be referenced throughout the paper where relevant. Of particular relevance is our work on the S0 and S1 agents, constructed in the Auto-catalytic Endogenous Reflective Architecture (AERA) framework (Nivel et al. 2013c), which demonstrated how the principles described herein can enable a machine learner to learn highly intricate real-time tasks by observation. S0 learned a complex goal-driven task (complying with commands provided in natural language and gesture) in only 2.5 minutes; S1 derived sufficiently detailed knowledge from observing humans in an interview about recycling various materials, also based on natural situated communication but with a much more complex 100-word vocabulary and free-form grammatical sentences. We give a short description of these agents in Section 6.

The paper is organized as follows. First we present a high-level overview of the theory, including introducing background concepts, with a particular view to ‘generality’ and ‘autonomy’ (Section 2). Then we will look at what kinds of worlds the present work is relevant to (Section 3). After this we look at the concept of seed-programming (Section 4). Then we discuss cumulative learning (Section 5), which contains four subsections, each focusing on

key aspects of such learning (modeling and semantic modularity, causal-relational³ models⁴ and micro-ampliative reasoning, respectively). We conclude with a summary (Section 6) of the preceding sections and a short conclusion (Section 7).

2. Autonomous Generality

We consider the effort to create systems with “general intelligence” to be about implementing systems with *autonomous generality*, since the requirement to handle an increasing variety of situations, environments, and worlds grows (i.e. increase in generality), increases the requirement on autonomy too, since a generally intelligent agent that cannot think independently, on its own – autonomously – is by definition not general.⁵ Conversely, to achieve high levels of autonomy in complex partially-observable worlds requires increasing levels of generality.

The concept of ‘generality’ in the context of intelligence can be addressed from various angles, a common one being in relation to variety of one sort or another that an intelligent agent can handle—kinds of tasks, data, situations, environments, domains, or ‘worlds’ (Thórisson and Helgason 2012). While it would be nice to have an absolute measure or scale of generality,⁶ such a scale is rather meaningless without a general theory that provides a rubric on which it rests (cf. Goertzel 2015). At the very least there should be some way to say that one AI is more general than another. This, however, may be even more challenging than would seem at first, because generality is a multi-dimensional concept.⁷

Assuming a capacity to *learn* as given in any system classified as intelligent, and generality to refer to an enumeration of variety, what then is autonomy? Autonomy has been associated with artificial intelligence for decades and has commonly made an appearance in robotics and multi-agent systems, where it has been analyzed both with respect to tasks and capabilities (cf. Beernaert et al. 2018) and cognitive capacity and architecture (cf. Thórisson and Helgason 2012, Wooldridge and Jennings 1995). While both practically and theoretically useful for comparing such systems, the reach of this work typically does not include the concerns of biologists, whose interest in understanding autonomy from a self-organizing auto-catalyzing perspective aims to uncover the principles of autonomy in nature, especially life itself (cf. Winning and Bechtel 2019, Letelier et al. 2011).

We may think of autonomy as spanning at least three grand levels of sophistication: First, the lowest level may be called the “mechanical” level of autonomy—here we would list the familiar Watt’s Governor and old-fashioned thermostats, as well as modern deep neural networks and the like. Their function is fixed after they leave the lab, which is why some feel hard-pressed to classify them as autonomous, preferring instead the term ‘auto-

3. We follow a pragmatic definition of causation, in accordance with the practical needs of practical intelligence, taking it to mean the useful coupling of being able to achieve an effect by taking action (cf. Pearl 2009, 2001).

4. The meaning of the term ‘model’ in our use is comparable to that used in cybernetics (cf. Ashby 1956), which coincides pretty well with the use of the term in the vernacular. This is further discussed in Section 2.

5. Thanks to Leonard Eberding for this angle (personal communication).

6. For a formal definition of the concept of ‘generality’ based on entropy, see Goertzel (2010).

7. For instance, is an agent that can learn many things but takes a very long time to do so more general than an agent that can do 1000 tasks extremely well but can hardly learn, or does learning always trump staticity?).

matic’ and ‘automation’ for systems at this level. They may nevertheless “perform complex tasks” unaided (before they break). At the middle level we put system with some cognitive autonomy: The ability to adapt and handle novelty, to “figure things out” and even create new ideas. At this level we see humans and higher-level animals including some dogs, crows, and parrots. The highest level is the biological one, where life springs forth and spawns the other two levels. It is in our view the “most autonomus” one – should we try to pit them against each other – because it is a prerequisite for the others. However, it is a distraction to those focusing on artificial intelligence because intelligence calls for discretionarily constrained adaptation: The ability of the system to constrain its own behavior from knowledge *by choice*, through selecting and generating goals, sub-goals, new knowledge, and other factors, at its own discretion, through reasoning and logic (Thórisson 2020). On the border between the mechanical and cognitive levels of autonomy lie (basic) reinforcement-based learners that, while able to change their function at runtime, are limited to a handful of predefined variables.

In this paper we look at autonomy from the perspective of knowledge representation and acquisition at the cognitive level—especially in relation to “figuring out new things” (we will discuss novelty in Sections 3.1). While quite possibly less challenging than life itself, it does share some of the same concerns (cf. Rocha 2010) and harkens back to the early times of cybernetics, in particular what has been called ‘second-order cybernetics’ (cf. Heylighen and Joslyn 2001). Some work in developmental robotics has also been focused on these subjects (cf. Smith and Gasser 2005, Weng 2004), human cognition (cf. Piaget 1953) and artificial general intelligence (cf. Wang et al. 2016, Rocha 2010, Wang 2009). Our concept of learning is fairly broad, incorporating not only the accumulation of (practical) knowledge but also the ability to use this knowledge achieve some *autonomy in learning*. Other things being equal, a system that is general and autonomous when *performing* tasks, but not autonomous in inventing or finding new solutions, will not be as generally autonomous as one can do both. If systems capable of cumulative learning can take initiative and actively uncover missing information, and even invent new concepts for the purpose of getting better, they are able to not only do things but also “figure new things out.” Yet, if they are not sufficiently general their autonomy will be bounded to a limited set of tasks, circumstances, data formats, etc.: All of these are necessary (but not necessarily sufficient) for general intelligence.

This, then, is what we wish to create: An autonomous general cumulative learner. Our theory of autonomous general learning rests on the hypotheses that learning is

- (a) organized around causal relations,
- (b) bootstrapped from observed correlations, using
- (c) fine-grain relational models, manipulated by
- (d) micro-ampliative reasoning processes.

We further hypothesize that

- (e) a machine properly constructed in this fashion will be capable of *seed-programmed autonomous generality*: The capacity to apply learning to any phenomenon without help – i.e. being autonomously domain-independent – provided that
- (f) its seed references observable variables (at “birth”) and that

- (g) existing knowledge shares one or more observable variables, patterns or inferred features with novel phenomena to be learned.

Of course, this begs the question what is meant here by “properly,” something that we aim to address in this paper. It should be noted that our use of the term ‘model’ does neither imply mathematical (axiomatic) models, nor any kind of logic based on model-theoretic semantics (Taski 1944, Putnam 1981); rather, like Wang’s experience-grounded semantics (EGS; Wang 2006) our models model the experience of an agent, in particular indications of causal relationships that may be found there. Unlike ‘models’ in artificial neural networks, from which they are also quite different, our models have internal semantics that carry down to their smallest components: patterns, made up of variables, transformation functions and relations.

To explain the background of our theory, three high-level things must be considered, as already mentioned: The world of the learning agent and its target *task-environments*; the mechanisms for control and management of the *cumulative learning* process, i.e. cognitive architecture and knowledge representation; and how learning is bootstrapped through the information it is born with—its *seed*.

Task-Environment. As argued by Steunebrink et al. (2016), no general intelligence in a complex environment such as the physical world can be granted access to a full set of axioms of the system it’s controlling, let alone the $\langle agent, environment \rangle$ tuple, and thus the behavior of a practical generally intelligent artificial agent as a whole simply cannot be captured formally. A key target environment of the present work is the physical world, which follows (local and global) rules but will always contain vastly more unobservable, un-accessible and unmanipulatable variables than those any learner – hypothetical or practical – will ever know. In such a world novelty abounds—most things that a learner encounters will contain some form of novelty. We look at this in Section 3, *Kinds of Worlds*. We assume that our physical world (“the real world”) is in fact hypothesized – induced from experience – because, as many philosophers have pointed out through the ages, all the evidence we have is our own experience of it. This viewpoint is important when constructing a general learner because it means we *must* infuse it with an ability to bootstrap learning from experience.

Seed. A learner that is born knowing nothing cannot learn anything, because there is nothing to tell it how (or what) to learn; like the proverbial baby that’s thrown into the deep end of the pool, it cannot learn to swim if it doesn’t even know how to move its arms. That natural intelligence is a blank slate at birth – a *tabula rasa* (Aristotle 350 BCE) – clearly cannot be correct. A seed must thus include *some assumptions* about the world the agent is born into, to bootstrap its learning—an “inductive bias” (Dubitzky et al. 2013) or *bootstrap program* that tells the newborn what to initially pay attention to, out of the myriad of possibilities, and how to turn experience into knowledge.

Representation. In a novelty-rich world, no challenge, problem or phenomenon is identical from moment to moment, if only for the simple reason that the progression of time moves things continually from the present to the past, changing the context: Time is a semantic property (Lee 2009). A cumulative learner figures things out because it is equipped with corrigible – non-axiomatic (Wang 2006), defeasible (Pollock 2010) – knowledge, and as it encounters novel problems it augments its current knowledge as well as modifies it to adapt the new information. Ashby’s Requisite Variety Theorem (1958) states that the representation of a controller must have a granularity as small as the smallest feature that we

want the learner to discern. Our approach uses a two-part representation scheme, consisting of causal-relational models (*CRMs*) and patterns—both of which are information structures amenable to manipulation, comparison, and compositionality. Equally importantly, they are abstract and can thus be used at any level of detail; their capacity for abstraction comes from their ability to form hierarchical sets. Key management mechanisms over this representation are (corrigible) abduction and (corrigible) deduction.

In prior work we have demonstrated an implemented architecture, the Autocatalytic Endogenous Reflective Architecture (AERA, Nivel et al. 2013c), that incorporates these ideas. AERA agents can learn autonomously many things in parallel, at multiple time scales, as experiments with the S1 and S0 agents show. Results of detailed evaluation of these agents show them capable of learning complex multi-dimensional tasks from observation, while provided only with a small seed including a simple ontology, a few drives (high-level goals), and a few initial models, from which it can autonomously bootstrap its own development (Nivel et al. 2014c). This is initial evidence that our constructivist methodology is a way for escaping the inherent allonomic constraints of standard computer science and engineering methodologies.⁸ Human dialogue is an excellent example of the kinds of complex tasks current systems are incapable of handling autonomously. The fact that no difference of any importance can be seen in the performance of AERA and humans in face-to-face interviews is an indication that the resulting architecture, and CAIM (Thórisson 2017, 2012), hold significant potential for further advances.

3. Kinds of Worlds

Without regularity, no patterns would repeat, no event would reliably result in something specific, no categories of objects or events would exist, and no action would lead to any predictable outcome. Such a world would consist entirely of noise and no learning could take place. On the other hand, completely deterministic, fully observable worlds call for little guesswork. Whether fully deterministic or not, regularity means that some things go together more than others.⁹ A world where some things reliably precede others at the exclusion of yet other ones is a *causal* world.

At the scale of physical reality where human intelligence operates – the *object* scale¹⁰ – a lot of regularity is found, allowing us to survive in complex environments. An intelligent agent in the physical world must deal with (1) partial observability, (2) an enormously large number of building blocks, from glare on water ponds to the faces of relatives to sunsets by

-
8. An allonomic system does not generate its own semantics, relying instead on the semantics provided by its creator (Thórisson 2017). While perfectly acceptable for normal engineering efforts, it preempts the requirement of artifact autonomy by requiring its semantics to be infused at design time, instead of being provided with methods that allow the artifact to develop its own semantics, autonomously.
 9. Whether the physical world is fundamentally deterministic or not “at its core” is a metaphysical question and immaterial for the present purposes—what matters here is that any agent aiming to survive in it must make do with the limited information it has, as stated by the assumption of limited knowledge and resources (AIKR, Wang 2006).
 10. The spatial lower bound of (unaided) sensation/perception is not much under 1 millimeter; the temporal lower bound is around 10 milliseconds. The lower spatial (unaided) resolution of the perception-action loop is a few millimeters and temporally around 100 msec. If we limit the upper bound to our maximum lifespan all those dimensions would max out at 100 years. The vast majority of human cognition and perception-action events span the range from 1 to 10⁴ seconds.

the beach, that (3) can combine in an exponentially greater number of ways over the lifetime of a learner. If we define ‘novelty’ as any pattern that may diverge in perceptually noticeable ways from patterns seen before, we can safely say that novelty is far more common than precise (imperceptibly different) repetition. According to Wang (2020), dealing with this is in fact a defining property of intelligence.

From a learning agent’s perspective, whose memory can only fit a minuscule fraction of the world’s combinatorics in memory (even if they were strictly limited to the object scale), any method or “trick” that makes its tasks easier is welcome. Three critically important features make intelligence in the physical world a practical possibility: Multi-level regularities, a.k.a. repeated patterns, and repeatable transition functions, a.k.a. “laws of physics”.¹¹ The third is the proverbial arrow of time, composing what we call “reality” out of temporally morphing patterns that follow lawful transition functions. While the ocean may never repeat the same waves exactly the same way twice, at a higher level of analysis its waves share sufficient similarities for meaningful comparison (and contrasted with other patterns in other contexts). For data measured at the lowest level of sensors (e.g. a 2-D grid such as the eye’s retina or a pixel sensor of a digital camera), this hierarchy of patterns makes what otherwise would present too much novelty to keep track of, in any practical manner, analyzable at higher levels in groups or blobs, dissecting a visual scene into what we know as trees, mountains, buildings, and people, each of which in turn can be seen as consisting of parts such as branches, roofs, ears, noses, etc. Similarities between spatio-temporal levels forms yet another dimension of similarity comparisons, presenting in essence a fractal dimension—yet another “trick” that intelligence can use to handle complexity. As has been pointed out before (Richardson 1998), the multitude of factors that matter to an autonomous agent in the physical world can be thought of as forming a *dynamic hypergraph*, where values of sets of particular variables dictate the values or value ranges of others (one need only think of the way that walls of a room limit the possible positions of objects within the room).

The world W consists of a set of variables \mathcal{V} , dynamics functions \mathcal{F} , an initial state \mathcal{S}_0 and relations \mathfrak{R} between these: $W = \langle \mathcal{V}, \mathcal{F}, \mathcal{S}_0, \mathfrak{R} \rangle$. The variables $\mathcal{V} = \{v_1, v_2, \dots, v_{\|\mathcal{V}\|}\}$ represent all the things that may change or hold a particular value in the world. The dynamics can intuitively be thought of as the world’s “laws of nature,” continually transforming the world’s current state into the next: $\mathcal{S}_{t+\delta} = \mathcal{F}(\mathcal{S}_t)$. The concept of ‘domains’ as subsets of the world, $\mathcal{D} \subset W$, where a particular bias of distributions of variables, values and ranges exists, may be useful in the context of tasks that can be systematically impacted by such a bias (e.g. gravity vs. zero-gravity). Each variable v may take on any value from the associated domain $d_v \in \mathcal{D}$; for physical domains we can take the domain of variables to be a subset of real numbers.

A phenomenon in the world is any grouping of variables and relations in the world that we choose to group as such; $\Phi = \langle \mathcal{V}_\Phi, \mathfrak{R}_\Phi | \mathcal{V}_\Phi \subseteq \mathcal{V}_W \wedge \mathfrak{R}_\Phi \subseteq \mathfrak{R}_W \rangle$. It consists of elements $\{\varphi_1 \dots \varphi_{\|\Phi\|} \in \Phi\}$, which can themselves consist of other phenomena, variables, and relations

11. Calling them “laws” is a bit misleading, philosophically speaking—to be certain of their status as *laws* would require a third-person view of the universe and our existence, something we can of course never attain. Nevertheless, a philosophical stance alone is insufficient for advancing the science of AI, for this we must rely on science, including physics; as a matter of practicality we feel justified in ascribing the source of our agents’ perceptual experiences to variables and physical patterns.

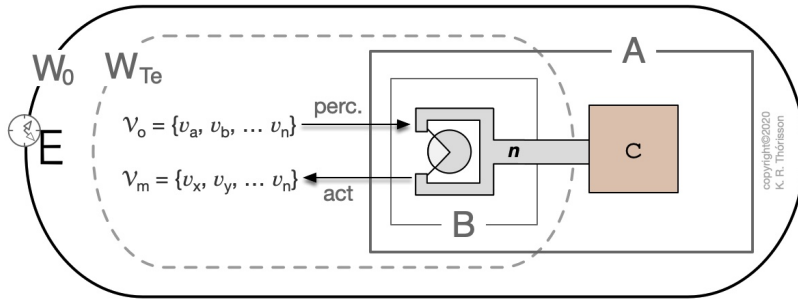


Figure 1: An agent A in a task-environment embedded in a world ($Te \subset W_0$) that is subject to time and energy constraints, consists of a controller c and a body B consisting of one or more (n) transducers that can observe accessible variables (\mathcal{V}_o) and affect manipulatable variables (\mathcal{V}_m), where $\|\mathcal{V}_o \cap \mathcal{V}_m\| \neq 0$. While the body belongs to Te we may think of the controller to be either outside of it or part of it (the latter will be true of any implemented controller). The agent acts by interacting with the environment (‘act’) and measures the effects of such acts (‘perc.’) through its transducers. Knowledge is thus constructed solely from data flowing to and from the transducers—the controller cannot say anything concrete about the “reality” of a “world outside itself” (the ‘exogeneous’ part in Figure 3), which permanently remains induced from (hypothesized based on) experience.

\mathfrak{R}_Φ (causal, mereological, etc.). \mathfrak{R}_Φ couples elements of Φ with each other, and with those of other phenomena in the world (Bieger and Thórisson 2017, Thórisson et al. 2016b), and can be partitioned into inward facing relations \mathfrak{R}_Φ^{in} between element pairs $\varphi_i, \varphi_j \in \Phi$ and outward facing relations \mathfrak{R}_Φ^{out} between element pairs $\varphi_i \in \Phi$ and $\beta_j \in W$.

An environment e is a view or perspective on the world, $e \subset W$; a *task-environment* Te is an environment where a set of tasks can be assigned and performed and where some variables are observable (\mathcal{V}_o) and manipulatable (\mathcal{V}_m)—parts of the world which may change dynamically as the state-space of the hypergraph it forms is traversed, based on relations between variables. The state-values of any element or variable in Te at any point in time is determined by a causal chain that we can think of as a directed acyclic graph. Any complex environment is one in which the total number of variables is vastly greater than the total number of observable variables $\|\mathcal{V}\| \gg \|\mathcal{V}_o\|$ and observables more numerous than manipulatables $\|\mathcal{V}_o\| > \|\mathcal{V}_m\|$, since these factors directly contribute to a system’s complexity from an agent’s limited view of the world.

3.1 Agents in Task-Environments

We define an *embodied cognitive system* as consisting of two main interacting components: A *goal-driven learning controller* of a body, and its task-environment.¹² By “body” we mean a set of transducers (sensors, actuators) that bridge between the world and the controller and

12. This formulation is commonly used in control theory and reinforcement learning, but is general enough to also cover more “narrow-AI” cases of supervised and unsupervised learning. It is also very convenient from an engineering perspective even though as already noted, we cannot really know what lies beyond our direct experience in the physical world. For simulated worlds, where we know *exactly* what it consists of and how it works, it is convenient to model it (in part or in whole) from a third-person (“god’s eye”) view. This does not change the fact that a designer of general machine intelligence must ultimately target the physical world, which contains unknown and novel phenomena.

specify how and what the agent measure and affect, at any point in time, in the world. By “goal” we mean a set of target variables and associated values, with permissible deviations and time limits, as well as potential constraints (“negative goals:” that which must be avoided while working on the task). By “goal-driven” we mean that the system actively seeks state spaces that meet certain conditions, as specified by its goals; by “learning” we mean that experience may be used to direct subsequent behavior in favor of such goal seeking, to do it better (e.g. fail less often upon repeated attempts, get closer to the target values of the goal with practice, or achieve the goal with less effort or time).

A basic task T in a particular environment Te is any specification that consists, at a minimum, of an initial state $\mathcal{S}_0 = \{v_1, v_2, \dots, v_n\}$ (or a set of them; variables in the environment having values within particular ranges) and time t_{go} , a top-level goal $\mathcal{G}_{top} = \{\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_n\}$ (a set of states with desirable joint variable assignments that must be achieved at particular times) and sub-goals, $\mathcal{G}_{sub} = \{g_1, g_2, \dots, g_n\}$ that serve to further specify \mathcal{G}_{top} , constraints \mathcal{G}^- (i.e. negative goals—states that result in failure), an interface to W that includes a set of transducers with a reasonably stable relationship to each other (i.e. a body) and accompanying affordances¹³ that we refer to as the agent’s body $B = \{b_1, b_2, \dots, b_n\}$, and co-located with the controller), an end time, t_{stop} , and possibly some additional information¹⁴ about the task, e.g. instructions I : $T = \langle \mathcal{S}_0, \mathcal{G}_{top}, \mathcal{G}_{sub}, \mathcal{G}^-, B, t_{go}, t_{stop}, I \rangle$. Task parameters may be specified in various ways, for instance, t_{go} may be given as a particular sub-state of Te , which then becomes a precondition for doing the task; however, t_x and other values must be bound for task performance to be possible. An *assigned* task will have all its variables bound and reference an agency that is to perform it (accepted assignments having their own timestamp t_{assign}). Complex tasks may be thought of as being composed of sub-tasks, $T_1 = \{T_a, T_b, \dots, T_n\}$, forming a web of relations through one or more shared variables.

When performing a task an agent may sample values of variables from the environment, produced by its transfer functions $\mathcal{S}_{t+\delta} = \mathcal{F}(\mathcal{S}_t)$, through measurements via its transducers; we say that a *percept* p is a subset of variable values, $p = v \subset \mathcal{V}$, that are measured by an agent’s transducers at some point in time (or over some period) and accessed by its controller (i.e. perceived), and that $\mathcal{P}_{\varphi(t)} = \{p_{1(t)}, p_{2(t)}, \dots, p_{n(t)}\}$ is a set of percepts related to one or more elements $\varphi \in \Phi$ at time t . For the kinds of task-environments we are interested in, the number of exposable variables at any point in time is typically vastly larger than observable ones, $\|\mathcal{V}_{e(t)}\| \gg \|\mathcal{V}_{o(t)}\|$. Only variables that are *exposable* can be observed, and only variables observable at time t can be part of a percept; thus, observability is a temporal property of variables but exposability is not. Exposable variables may be unobservable for various reasons and at various times (most commonly because they are obscured by objects or because they simply are not in view of the agent’s transducers, but also if they require a particular technology to be invented to be seen, e.g. a microscope.) We may assume that

13. “Affordances” (Gibson 1966), in our conceptualization, are quite simply *familiar verified causal chains* relevant to any state-goals tuple, $\langle \mathcal{S}, \mathcal{G}, \mathcal{G}^- \rangle$. In the case of transducers this involves the primitive data that can flow in and out of them (input and output from the controller).

14. ‘ T ’ refers to ‘instructions’ and other relevant information—anything that must be conveyed to the agent about tasks so that it can learn and/or perform it. A reinforcement learner receives “instructions” in this sense via a set of pre-selected variables (“these are the variables that matter, and this is how they are related”) and a time-series of good-bad reward signal for its actions. The fewer constraints a learner has in the type and form of instructions it can accept and make use of, the more powerful and flexible it will be.

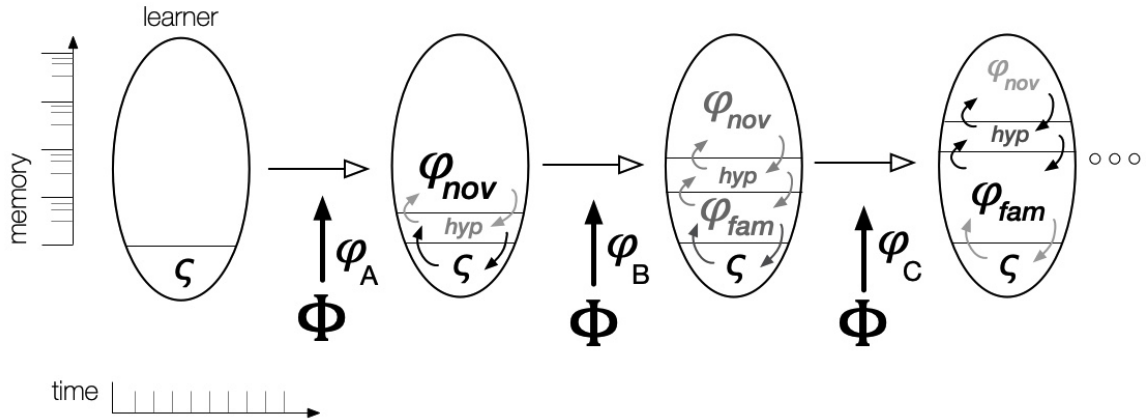


Figure 2: The amount of knowledge an implemented learner can hold (represented by an oval) is always bounded by physical limitations. At the outset its only knowledge is contained in its seed (ς); everything in the world is completely novel (Φ_{nov}) except those percepts that are referenced in part, in some way, by information provided in the seed. As a learner encounters percepts of a novel phenomenon Φ over time, $\varphi_A \in \Phi, \varphi_B \in \Phi, \dots, \varphi_n \in \Phi_n$, it creates models of what it observes, which initially are hypotheses that may get strengthened through experience, reasoning, and their fit with existing knowledge. The hypotheses are bootstrapped after “birth” (second oval from left) via interaction between information gathered through experience and information in the seed (ς), and later by newer knowledge, Φ_{fam} . Over time, the amount of its percepts related to aspects of phenomena it can predict (Φ_{fam}) increases; reciprocally, the amount of aspects of the phenomenon that are novel (Φ_{nov}) decreases. The seed (ς) typically does not change: If its drives were to change it would risk unpredicted learning and development; if the models in the seed change it is typically because they are not needed any more or because new, better ones have been learned. The seed may contain information structures for several levels of its agent’s development (“level” in the sense of Piaget (1976)), only to be triggered when certain developmental stages are reached. As knowledge builds up the knowledge management mechanisms must rely on compaction methods to add knowledge to memory (cf. Thórisson et al. 2019). This will involve some forms of induction and forgetting, among other methods.

percepts are always in the “now”; a “percept” that is not being delivered directly by a sensor modality ‘right now’ is not a percept but rather an encoding or representation – a memory – of a percept. The minimal duration over which this may happen could be defined as the smallest temporal difference that the controller can handle.

Hidden variables may become visible, and others may become hidden, as an agent interacts with phenomena in the world, exposing their relationship through their transition functions via systematic state comparisons of time periods. Any phenomenon that a learner may encounter, and of which it has no knowledge, is by definition novel to the agent, Φ_{nov} . As an agent learns, that novelty turns into familiar information, Φ_{fam} , and elements of Φ cease to be novel, $\Phi_{fam} = \Phi - \Phi_{nov}$, as knowledge grows over time. We will come back to this shortly. In the kinds of task-environments we are interested in, for any agent, the set of novelty is vastly larger than that of what is familiar, $\|\Phi_{nov}\| \gg \|\Phi_{fam}\|$, even after lifelong learning. It is therefore given, in the case of the agenthood we are interested in, that the set of unknown phenomena, at all times, is huge.

4. The Need for a Seed: Seed-Programmed Autonomy

Human knowledge of the natural world – indeed, that of any learner in the physical world – is the result of a composition process, pieced together incrementally from experience with the world over time, accumulating in a somewhat systematic way. This is the way any learner must operate where complete information is not available at birth. If an agent is to learn independently, autonomously, its knowledge acquisition processes must be self-guided—it must have *existential autonomy*. To successfully engineer artificial systems with this capacity requires uncovering principles of such *cumulative learning*.

If we wish to create an autonomous learner that can operate without human intervention, or other kind of such outside interference, reprogramming etc., the system will by definition have to be “born” with a set of initial knowledge—a *seed* from which the learner’s knowledge should grow cumulatively, as it interacts with the world, in a self-supervised fashion. If we define a “newborn” cognitive agent as one with only a seed and no experience of interacting with the world, a newborn’s first task is to bootstrap its own knowledge acquisition process, starting from what’s in its seed. What kind of information must exist in such a seed? If we are designing a system from scratch it will be up to us to decide; however, if the aim is to make the system autonomous, the information must be sufficient for the system to bootstrap its knowledge; our hands are tied after we bring it into the world, so we must choose wisely...

The purpose of intelligence is to get new stuff done. For natural intelligences this inevitably involves surviving and making offspring (lest the species die out); for an artificial agent the main purpose – we call this top-level goal, or preferably *drive*¹⁵ – may be to manufacture computer parts, design new manufacturing plants, or invent new ways to compute. Intelligence is a practical solution to practical problems, and few things are as practical as *taking action*,¹⁶ to get something done, preventing something from happening or affecting the world in some way—through deliberate achievement of *goals*. If intelligence is not good for taking some sort of action in a time of need, what is it good for? (Absolutely nothing.)

What kind of action should a seed enable a newborn to take? A learning agent’s motivation for acquiring and improving its own knowledge – its impetus for modeling the world – must exist at birth, otherwise no learning will take place. To work, such motivation must reference actionables and observables in the world, even if it does so indirectly. This leads to the following principle:

§1 *The seed of an autonomous general learner must include at least one (top-level) drive that references (sensory) variables necessary for grounding that drive.*

By “grounding” we mean the ability to build models that have the potential to operationalize the drive, making it practical. A seed for a non-axiomatic task-environment, such as the physical world, will inevitably be something that *might* work for bootstrapping the learning

15. While technically similar to goals, drives deserve their own name because in a general learner they serve a different role than a typical task goal or sub-goal: They ensure the learner’s *purpose*, and are likely to be implemented in a somewhat different, possibly more rigid way. A drive does not have any ‘why’ related to the agent’s actions, like top-level goals and sub-goals, it only refers to the species’ nature or a machine’s purpose.

16. In a world with a real-time clock, inaction also has consequences and may be used to get things done (or messing them up).

and development—a kind of “suggestions for things to test,” because the world into which a being in the natural world is born cannot be fully known. This also means that the seed and the transducers must be, to some level, coordinated.

For an agent constructed to do more than one task – let’s just say various household chores, for the sake of argument – could we not provide it with a detailed description of every such possible task up front—dusting, vacuum cleaning, loading and unloading the washing machine, etc.? If we had to do that its seed would have to contain an impractical amount of detail (think of the all the ways any even just a tiny portion of a lacquered floor may look, from different angles, under different lighting conditions, throughout the year). That is not the only reason though for this being impractical; such a seed – if we could pull it off – would be very sensitive to minor changes in any task and environmental variables; anything that is slightly different from the seed would stop the agent from being able to achieve it successfully. Instead, the seed should contain generalizations in the form of rules. (We will discuss the form they may take shortly.) The main reason for populating the seed with general rules is that not everything may be known up front—even the set of tasks that we might want the agent to perform may change drastically in the future. In a complex world with infinite combinatorics, this is eventually bound to happen: Novelty is guaranteed and thus must be assumed when constructing a seed.

We have uncovered a general and fundamental principle of general intelligence: A general learner must be able to address novelty (i.e. map the novelty in light of Φ_{fam} and relate it to its agency). If the seed is to be of use in bootstrapping a *general learner* it must appeal to some *general principles* of the world into which it is born. The more general we want an agent to be, the more general the initial rules at birth must be—remember, at birth the seed contains *the entirety of the agent’s knowledge*. The more specific the rules are, the more sensitive they are to future changes in conditions, tasks, and environments. Could we start the general learner with an empty seed? No, because the purpose for intelligence is to control action, and the learning that the seed is supposed to bootstrap must eventually lead to a *discovery and verification of ways of affecting the world*. A seed must thus contain some instructions precisely about that – affecting the world – in particular, how *this particular agent* can affect the world via *its own body*, which is its interface to the world (physical forces), in these particular circumstances. The seed code must reference parameters that are assumed to be actionable in the learning domain upon initial bootstrapping, through manipulatable variables (e.g. outer limbs and movable sensors). This is necessary but not sufficient: To know whether one action is good and another one bad, the seed must be populated with some general reference to observable variables—measurable entities whose existence in the birth world is fairly certain; variables that can affect the agent’s sensory aparati. Lastly, the seed must supply some ways in which actions and observations may be associated or related in an information-crowded world. Additionally, for a learner in a complex environment, the seed may include information structures for several levels of development, only to be accessed or activated when certain milestones of cognitive ability have been reached (Piaget 1976), which requires some form of introspection. The seed (ς) typically does not change, however—if its drives change there is a risk of unpredictable behavior and development; if its models change it is typically because they are not needed any longer or because new, better ones have been learned. As knowledge builds up the

knowledge management mechanisms must increasingly rely on compaction (compression) methods to fit more knowledge into the limited memory (cf. Thórisson et al. 2019).

In the physical world, the most fundamental principles known are the laws of physics, as they have slowly been uncovered in fits and spurts over the past 2000 years. One problem with the laws of physics, even though many of them may be represented compactly, is that they are very general. In science that’s a feature of course, not a bug, but in the context of a seed it means that they do not specifically reference anything that the agent can directly sense or act on (that’s in part why it took so long to uncover them). Humans were not born with knowledge of these—they had to discover/invent them. A learning agent in the physical world will not only be learning sub-divisions of what it sees and hears, it will be learning about how things change over time. Providing rules about relations between action and perception is difficult without using complex concepts—concepts that most certainly will be beyond a newborn’s ability to grasp. The agent learns about the world little by little, over time, through its experience, but what it can perceive at the outset may turn out to be rather limited.

The relations that make the biggest difference in a (partially, supposedly) lawful world, yet are compatible with cumulative learning, are the rules that allow a particular action to produce a particular effect in particular conditions. Anything that reliably can be done to affect the world – with primitive actions, by the way, because there are limits to how much sophistication should and can be stored in a seed – could serve as a starting point for learning about the world. Learning to achieve goals in a complex world can only be achieved by causing something to happen, but to cause something specific to happen – out of an infinity of alternative options – means we must know about causal relations. Causal relations capture in a compact way a principle of how things come to be: Via interactions between demarcated phenomena, where one begins and triggers another. This is the most fundamental assumption that any general learner can make: Assumption of the existence of causal relations—*AECR*:

§2 *Causal relations must be a fundamental organizing principle of general knowledge representation.*

Others have argued the same (cf. Pearl 2019, Pearl 2012) from different angles, yet for similar reasons. Note that we do not have to answer any philosophical questions about whether causes and effects “really exist” or whether they only exist from the viewpoint of the observer—this is simply the name we give the observation that in a lawful world, even a *partially* lawful one, not only are some things better predictors of particular outcomes than others, some are better for producing particular outcomes through *volition*. In such worlds agents have a choice between options. It is a practical definition, fitting for a practical solution to practical problems (i.e. intelligence). From this practical stance we see causal potential a gradient, meaning that some such relations are better than other for various purposes. However, in this view time is nevertheless an absolute and prime differentiator: Effects cannot happen before causes; any good evidence to the contrary will automatically disqualify a hypothesized cause as such. Seeking out rules that capture stable cause-effect relations, and seeking to replace bad ones with better ones, are thus at the center of any agent that wants to get things done.

If an agent is to learn some aspects of perception, we might want to put some general (meta) rules about extracting patterns from the world (this is in effect what nature does in

many species; it is a mapping from predicted world stimuli to the agent’s percepts—more on this later). These, too, should be made as general as possible for a general learner, but on the downside it limits the range of other rules we may put in the seed because to learn, the agent must be able to perceive from the outset. A potential solution is to include a *development program* for the perception-cognition-learning tuple in the seed (this might be the source of developmental stages as observed in natural intelligences).

Another conclusion from this is that the seed must make *some assumptions* about the world; it cannot offer a completely blank slate – a tabula rasa – because without reference to *something observable* in the world, the newborn learner has nothing to grab onto.

§3 *The seed of a newborn autonomous general learner must reference some variables that it can affect and sense.*

When a particular relationship is experienced an autonomous learner cannot simply “call home”—it must be able to interpret the *meaning* (to itself, and by extension to other things) of such relationships on its own; it must be able to build its own semantics. From these beginnings, models of causal (and other) relations would be built (Figure 2). However, this grounding cannot proceed unless there is (some) causal relationship between the seed variables (actions and observables).

If this were all there is to the story we would simply stop here. But of course it isn’t. To work with causal relations through cumulative learning requires mechanisms to deal with the passage of time, a potentially exponentially growing set of rules generated from a broad range of experiences, an inevitably increasing set of contradicting information, and a host of other issues. Let’s see how much further we can take this.

5. Cumulative Learning

Any controller that consistently, effectively, and efficiently achieves its goals is a ‘good controller’.¹⁷ As shown by Conant and Ashby’s *Good Regulator* theorem (1970), every good controller of a system must contain a model of that system. Models, in our approach, are autonomously produced via a process that relies on contrasting prior knowledge (or seed) and percepts of observed variables and their relations in the world, supplied from experience. This process is really an semi-experimental empirical process, equivalent to the scientific method (which is also model-based (cf. Dellsén 2018)). Any such process, in an intricate task-environment, faulty models are inevitable. To evaluate and verify models, they are measured on their *usefulness* for achieving goals: The more accurately, effectively and efficiently they help the controller achieve its goals and sub-goals, the more useful they are. A cumulative learner is thus also a cumulative modeler, and its capacity to learn lies in its combined ability to create models and use them.

Following Thórisson et al. (2019), we consider a cumulative learner a *learning controller* that is constantly encountering novelty that it must *unify* with its current knowledge—it is an *experience-based* learner (Steunebrink et al. 2016, Wang 2006). The learner’s behavior, including its learning, is guided by one or more top-level drives (a drive hierarchy), that results in regularities being extracted recursively from its experience – of self and environment

17. Points of reference for these adverbs can more or less come from any source; in nature some obviously come from starvation, death, and procreation.

– to construct unified knowledge useful for achieving goals in that environment (Thórisson and Talbot 2018b, Castellfranchi 2009).

Goals of varying complexity may be assigned to a goal-driven learner, by itself or someone else, composed of a set of sub-goals, each possibly involving a relatively large set of variables spanning potentially long periods of time, whose successful achievement requires diligent tracking of time, at multiple orders of magnitude. Goals, relations, variables, and other aspects of the world are represented as a growing network of (micro-) models.¹⁸ Cumulative learning is a process of *model unification*:

New information enters by default into a process of being integrated with already-acquired knowledge—whether it is in agreement with it or not. This is compression under requirements of incrementality, realtime, and generalization: Frequently improving current knowledge with demonstrably better knowledge and extending missing knowledge, while generalizing when possible, prepares knowledge to be efficiently applicable to the largest possible class of situations, tasks, topics, and domains—as soon as possible during the learner’s lifetime. (Thórisson et al. 2019, p. 199)

5.1 Cumulative Modeling

When brought into an unknown world with only a bootstrapping seed referencing a handful of variables that may or may not be causally related in those particular circumstances, one must stay alert to indicators of causal relations. In the “worst case scenario”—when a set of percepts seem as close to completely novel as they get—the only resort is working from *correlations*: The co-occurrence of events. Of course, correlation does not mean causation, but no causation is without correlation, and as long as some relevant variables are observable this is the most general way to bootstrap world modelling from meager beginnings.

§4 *An autonomous general learner bootstraps its knowledge about novelty from observed correlation.*

Causal-relational models (*CRMs*) may be created when the controller observes an event α and a subsequent event β that follows. The model can be seen as a hypothesis that the observed event α caused the observed event β , so that when observing again an event α in the future, this model will predict that β will be observed. Models that do this prediction better than others are kept and used, others are deleted. When a better model comes along it will be preferred over the old one(s).

Large dynamic state spaces (such as the physical world) will of course present, for any period of time of interest to a learner, a large amount of semi-co-occurring events. There will be a lot of correlation whose vast majority will be irrelevant or spurious. To limit the search space for interesting relationships a practical rule of thumb is to shrink down the spatio-temporal scope of such a search (it is no coincidence that Pavlov’s dogs salivate more when the bell during training is placed closer to feeding time, following specific spatio-temporal curves (Pavlov 1902). Other more powerful methods can additionally be used, which we will describe in Section 5.4.

18. In prior work we have referred to such models as ‘peewee-granular’ models (Thórisson 2012, Thórisson and Nivel 2009).

A cumulative learner’s successful unification of new models with its current knowledge is based on using current knowledge as a kind of seed, whereby new models are contrasted with current knowledge so that they may help highlight similarities and differences, in a process of comparison. A newborn with little knowledge (just its seed, ς), and primitive acting and sensing abilities, may be facing the most extremely constrained knowledge acquisition situation in its lifetime. Yet the two situations are identical in that an effective seed, as opposed to an ineffective one, must help a learner with *new information*. That is in fact what *any* useful knowledge will do – according to the canonical definition of learning – and therefore we consider

§5 *experience-based seed-programmed bootstrapping a special case of general cumulative learning.*

For all intents and purposes, this approach to modelling applies equally to seed-based programming as to model-driven cumulative learning in general. If the new information is only a small missing detail in an existing fairly complete model set (i.e. the agent already knows a lot about the phenomenon that the detail belongs to), unification may be straight forward because there are no conflicts with existing models and the new information simply “snaps in place” based on shared patterns. This is how learning something new about very familiar things will proceed in general. The difference between two tasks, $f_{\Delta}(task_1, task_2)$, for an agent may be measured by the agent’s ability to predict things in that domain.

For modeling the world around it, an autonomous learner in the physical world is not only limited to its own exclusive experience, it is also limited to *incremental* information collection. To make new information available as soon as possible for action and planning, and to avoid (spatio-temporal) *semantic siloing*, unification should happen as soon as possible during knowledge acquisition. If the information “chunks” to be unified are too big, siloing can happen due to a combinatorial explosion, making unification in essence impractical, resulting in either partial or improper unification, or complete abandonment of the effort. Part of the difficulty lies in the inevitable verification of the unification in an uncertain environment, where computational power, time and energy are limited and axiomatic proof is categorically impossible.

A practical alternative is *bounded* recursive self-improvement (Nivel et al. 2013b) whereby unification is accomplished by frequent and small increments, relative to the learner’s lifetime. This is as obvious in the simplest of real-world cases as it is in more involved ones, whether it is when we quicken our pace when we’re about to miss the bus, or start to save money a year before our scheduled expensive world cruise. By extension, the models should also be semantically simple, i.e. have few parts with simple operational principles.¹⁹

As discussed by Steunebrink et al. (2016), due to semantic (spatio-temporal) siloing, self-modifications must be fine-grained, tentative, additive, reversible (supports “undo”), and rated over time as experience accumulates—concurrently with all other activities of the system. The relatively simpler semantics of comparing small “snippets” of experience to existing knowledge than large ones means unification becomes more manageable. This is essential in worlds with limited time and energy, *LTE*. The comparison proceeds by isolating

19. This has the added benefit of making model behavior during processing highly predictable with respect to time, memory, and other resources, which is critical for recursive planning (where planning time is taken into account when planning (Nivel and Thórisson 2013b, Nivel and Thórisson 2008)).

similarities and differences (‘micro-analogies’), that allow new percepts to be dissected into parts and their relations. Because such dissection involves both static and dynamic properties, the process is a multi-dimensional comparison of dynamic models. An intermediate result of this process is a set of relevant known related knowledge, on which hypotheses can be based, that direct exploration of their implications using deduction-based simulations over established knowledge (good models). When simulated implications contradict current knowledge, choices must be made for how to resolve them, through a mixture of (micro-) reasoning methods.

One challenge a cognitive controller faces is finding the best balance between identifying and retaining (in memory) immediately relevant information – variables that matter to the tasks at hand – and other related variables that might interfere or help with tasks/goals in the future. Spending a lot of time on identifying and modeling apparently relevant variables may end up being wasted time, should those models never be needed. Repeated experience with relevant variables should enable a good learner to make good models, even in environments with a large range of variables spanning broad variability, enabling increased accuracy in prediction and goal achievement. The proficiency with which experience on separate instances of tasks undertaken allows good models to be produced determines in part how good a learner an agent is. As the quality of models increases, the frequency of surprises are reduced, and controller performance compromised by probability factors is lowered.

5.2 Semantic Modularity & Operational Closure

From the initial creation of new knowledge, and throughout its subsequent usage, modification, expansion, unification, and deletion, an agent’s learning mechanisms must be capable of self-supervised “surgical” operation on existing knowledge, as new evidence comes to light through experience. The information that makes up such an agent’s knowledge set must be structured in a way that supports reflective processes, including discrimination, comparison, and manipulation of *arbitrary subsets of the knowledge set*, so that over-generalization may be reined in, regularities shared by disjoint phenomena unified, recurring patterns generalized beyond their instances, new relations \mathfrak{R} created, deleted and changed, counterevidence weighed, conflicts ironed out, alternative outcomes and counterfactuals considered, and so on.

This kind of incremental targeted knowledge construction requires that the system knows (a) *what* existing knowledge the new information is *relevant* to, (b) *how* relevant it is to that knowledge, and (c) how best to *unify it*. These operations must be achieved through comparison, for which we have proposed a similarity comparison operator (Sheikhlar et al. 2020). The new information must be represented in a format compatible with the existing knowledge, in small and transparent ways so that the comparison can be mostly localized to relevant data, lest such comparison becomes impractically complex and computationally expensive.

Knowledge representation that meets these requirements is *semantically modular*, meaning that subsets of the knowledge carry white-box spatio-temporal semantics, such that that the application, relevance, and structure of the knowledge is transversally retained when dissected. This semantic modularity must carry down to the smallest knowledge “nugget” that the system intends to operate on when unifying new knowledge with old. Otherwise

the knowledge becomes *siloed*, building up over time as separate, loosely or entirely unconnected “buckets” of knowledge, each bucket being relevant to a particular subset of the task-environments the learner has encountered. The relevance and application of any particular set of information in a knowledge base depends in part of its context, represented by adjacent, related information structures, which in essence define its semantics (think of e.g. whether turning right on a red light depends on the state and country you’re in). In siloed knowledge the contextual information structures become “fused,” i.e. inseparable. When opportunity arises to update such siloed knowledge, each bucket – being independent of other ones – can only be updated by including all the relevant contextual information for that bucket, which inevitably is duplicated (in similar or identical form) in other silos, yet cannot be referenced due to its fused localized semantics. In such a setup, the ensurance and maintenance of the validity of prior knowledge during knowledge manipulation becomes a serious challenge. Processes necessary for cumulative learning become increasingly difficult to do the larger the silos are. The worst-case scenario is a single silo with no semantic modularity, which must be updated in its entirety when new information seems relevant, no matter how small. Temporary, experimental testing of alternative information structures is also prohibited, so the learner is limited in trying out different things. In addition to being problematic for seamless knowledge representation, silos should be avoided because they present a heightened risk of regression errors, resource shortage, and incompatibility with original top-level goals. Ashby’s *Principle of Requisite Variety* (1958), “AI’s corollary to Nyquist’s Sampling Theorem,” states that the resolution of a system’s representation must be at least as fine-grain as the smallest detail that it intends to handle. It lies at the core of our argument for semantic modularity and transparency.

Whether a *CRM* set refers to an atomic (relatively) well-defined detail such as the color of a jacket or a compound loosely-defined concept like “working in Iceland,” it must contain information that uniquely demarcates its target reference. This can be done by a particular and unique pattern of relations to other models, with which it shares some patterns, which in turn requires a similarity comparator (like the Ψ operator in Sheikhlar et al. 2020). Rather than forming a rigid structure, like classical ontologies (cf. Chandrasekaran et al. 1999), such patterns may be computed on an event-driven just-in-time basis, based on the current situation (as grounded through percepts—world situatedness) and active goals (as grounded in drives—cognitive situatedness). Any atomic concept, many of which are closely tied to basic transducer function, e.g. color, may in the simplest case require only a few models to represent; any complex concept will consist of a network of models, which again, are dynamically created based on the two “grounding sides” of the *cognitive now*: Situatedness – the perceived ‘here and now’ – and drives, whose derived active top-level pragmatic goals steer thinking.²⁰

As others have pointed out (cf. Wang 2006), a system that learns from experience in a constructivist manner – i.e. constructs its own knowledge from experience – must build its

20. Because our *CRMs* are fine-grain, any complex invariant in the physical world that needs to be computed – e.g. the apparent color permanence of a berry in in direct sunlight and in the shadow (which will register differently in the transducers but which the controller will want to perceive as being the same) – may require a large set of *CRMs*. Not all relevant models for such computation, however, need to be used on every occasion—this will be determined dynamically by context, facilitated by similarity measures and spreading activation.

mental constructs on a meaningful relationship between thought and effect, and this, we have already laid down some arguments for, boils down to modeling cause and effect. By “effect” we mean changes in variables measurable by a controller’s transducers, but also traceable internal events accessible to the controller’s monitoring of its own operation. The meaning of any mental operations – its *operational semantics* – is thus defined by its relation to monitored, operationally accessible and semantically discernible events. To be of any use for learning (to get better at targeted tasks, as well as getting better at getting better), these operational semantics must be *penetrable* by the system itself—their mechanisms must fit with cumulative modeling and learning mechanisms available to the controller. The cyberneticians called this “closure” (Ashby 1956). For cognitive development to achieve semantic and operational closure, *reflection* is needed.

We followed these principles directly in the implementation of the Autocatalytic Endogenous Reflective Architecture (AERA; Nivel et al. 2013c). For mental operations to achieve semantic and operational closure in AERA, *CRMs* are auto-generated and evaluated in-situ every time they are used, by applying uniform information manipulation processes that apply equally to transducer-originated information and internal events. All such processes are achieved through event-driven auto-catalytic principles. The methods and conditions for generating *CRMs* are a whole topic in and of itself; since new ones are essentially only hypotheses of “how the world hangs together,” and the system managing and using them can make use of massive parallelism, it is possible to generate a lot of them and simply try them out. This strategy only works, however, up to the point where the world starts to threaten the achievement of active goals, in which case more strategic methods of resource management are required. In AERA we have implemented a method whereby computations and time horizon for prediction can be automatically tuned based on computational load. Whichever attentional methods are available to the controller, however, its computational load will generally be highest when facing novel situations and tasks, which is why extracting useful causal relations as soon as possible for every situation will always be a central concern of any autonomous learner. As AERA is capable of reflection, such internal control falls naturally within its ability—all it needs is access to the relevant internal operations and parameters, and a seed (meta-)goal to “always achieve your goals,” to use its model-based learning methods to create the appropriate *CRMs* for when and how to control that.

5.3 Causal-Relational Models

An atomic model, in our approach, is based on relationships between two patterns, one describing prior condition and the other a subsequent condition (Nivel et al. 2013b). The prior state we say is on the left-hand-side (LHS) of the model, the subsequent on the right (RHS). Such models capture transformations of situations (states, events, conditions) which are represented as patterns referenced by variables, states, or other models. Read from left to right, a model can act as a predictor that says “when I see the LHS pattern I hypothesize RHS will come next.” These models are fully temporally contextualized and predictions come with a time attached; they are also semantically contextualized through relations to other models. The models not only hold patterns on their two sides, they also hold rules for how to compute an RHS state from a LHS, in forward chaining, and what may have

produced a given RHS (i.e. the LHS), in backward chaining. They contain specifications for their temporal constraints and how their variables are to be bound when they become relevant. When a model is considered relevant it is instantiated; any usage of a *CRM* in a particular interval is timestamped for that interval; the state and result of the particular instantiation in that context serves as a record of its usefulness. Because *CRMs* in our approach are peewee-size, a lot of them is generally needed to predict, act on, describe, and explain, a complex phenomenon.

Models \mathcal{M}_Φ is a set containing models of a phenomenon Φ , $\{m_1 \dots m_{\|\mathcal{M}_\Phi\|} \in \mathcal{M}_\Phi\}$. The closer the information structures $m_i \in \mathcal{M}_\Phi$ represent elements (sub-parts)²¹ $\varphi \in \Phi$, at any level of detail, including their couplings \mathfrak{R}_Φ , the better a system with models \mathcal{M}_Φ can predict Φ . For a model created from scratch based on correlation between two patterns, α and β , a strict requirement is that any pattern assumed for the LHS must meet a condition of having be measured to exist *prior to* the RHS, $\alpha(t_1) < \beta(t_2)$. This is one way to have the bootstrapping of learning start out honing in on causal relations from the very beginning. A model m 's *reliability* can be assessed at any time t :

$$(m, t) = \frac{e^+(m, t)}{e(m, t) + 1} \quad (1)$$

where $e^+(m, t)$ is the number of successful predictions produced by m and $e(m, t)$ the total number of predictions, both updated each time a prediction fails or succeeds (Nivel et al. 2015). A model's predicted usefulness at any point in time depends in part on this measure.

In addition to be used for prediction, models can be used for planning, because they are bi-directional (cf. Thórisson and Talbot 2018b), consisting of a pair of unified forward-inverse models (Wolpert et al. 2017, Castellfranchi 2009, Wolpert et al. 2007, Wolpert and Kawato 1998). The very same *CRM* that predicted “if α then β ” when read left to right, should say when read the other way, “if I want the RHS pattern β , I should look for the LHS pattern α .” This means that LHS patterns function as sub-goals, making every LHS in a chain of models a potential step in a plan.

The *context* of a phenomenon Φ is defined by its outward relations, \mathfrak{R}_Φ^{out} . For any complex phenomenon in a complex world, completeness of \mathfrak{R}_Φ^{out} is generally not to be expected, as this may be an extraordinarily large number. However, for any two phenomena Φ_1 and Φ_2 that are related, if $\|\mathfrak{R}_{\Phi_1}^{out} \cap \mathfrak{R}_{\Phi_2}^{out}\| = \textit{small}$, then predicting $\mathfrak{R}_{\Phi_1}^{out}$ may not require a lot of models of Φ_2 , even if $\|\mathfrak{R}_{\Phi_2}^{in}\| = \textit{large}$. An agent whose models are only accurate for \mathfrak{R}_Φ^{in} can predict Φ 's behavior in “isolation” but not how it interacts with other phenomena; if models are only accurate for \mathfrak{R}_Φ^{out} the agent can predict Φ 's relation to other phenomena, and thus how it interacts with them, but will not be able to predict the behavior of Φ 's internals, forcing a black-box view.

For compressing and improving any model set (metamodel) \mathcal{M}_Φ of a phenomenon Φ , the following procedure may be done at any time:

- A *Run a subset of the metamodel on a particular question and/or goal to produce predictions about that phenomenon.*

21. By “elements” and “sub-parts” we mean any sub-division of Φ , including sub-structures, component processes, whole-part relations, causal relations, etc.

- B** *Identify conflicting predictions.*
- C** *Produce hypotheses for the conflicts.*
- D** *For each conflict, produce one or more (inexpensive, easy-to-do) action(s) \mathcal{A} which has potential to invalidate one or more models related to the conflict, in accordance with the hypotheses.*
- E** *Sort the actions \mathcal{A} according to how many models they may potentially invalidate, descending.*
- F** *Next time the opportunity arises to act towards the phenomenon, select the top action and perform it (alongside, or instead of, other potentially relevant ones in the current context).*

This procedure effectively implements a mini-empirical experiment per cogitandi that helps prune semi-correct models and improves a metamodel \mathcal{M}_Φ related to a particular phenomenon Φ for predicting, achieving goals, and explaining Φ (some would say \mathcal{M}_Φ is moved closer to a hypothesized “ground truth”).

By making the *CRMs* hold variables instead of values, a generalized *model* is created whereby a range of future occurrences – with different but similar and/or comparable values – can be addressed by the model: For any instance where a model is considered relevant the model is instantiated with each of its variables bound to observed values. This means that induction is performed at runtime at the same peewee level of granularity as other knowledge processing events, as a natural part of the runtime of the system.

How *CRMs* are created from observation is not obvious; in the next section we describe in part how our combined abduction-deduction methodology removes less causal models based on correlation out of the metamodel \mathcal{M} , increasing its ratio of useful models over time.

5.4 Micro-Ampliative Reasoning

As we have seen, predictions through deduction are a central method for achieving self-supervised learning in our theory: Using available *CRMs* to anticipate what lies ahead.²² While prediction with *CRMs* is similar to the logical implication (\Rightarrow) in that in the *perfect* case a model that states $\alpha \Rightarrow \beta$, i.e. that if you see α you *will* see β , will always be true. However, *CRMs* model non-axiomatic data, based on the experience of an agent over time (e.g. in the physical world). Our deduction therefore cannot be classical axiomatic deduction, because the physical world’s axioms will be forever out of reach; the best a learner in this situation can do is hope that what it has come up with so far in terms of generalizations, is *good enough*. This means that even if we had a *CRM* m_1 that says $\alpha \rightarrow \beta$, and experience tells us this always to be the case, there is still no guarantee that α really is the cause, because that is only based on our experience *so far*. (They are therefore rightfully considered *pragmatic hypotheses* of causal relations, standing as potential indicators

22. It should be noted that rarely is deduction-based prediction “pure,” i.e. reliant only on experienced and verified knowledge, because it often involves a set of unknowns, demanding “placeholder” knowledge be created (most likely on demand) via induction, analogy and/or abduction, resulting in what typically is referred to as “assumptions.”

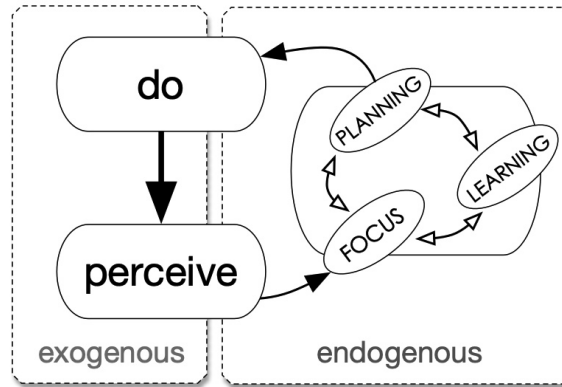


Figure 3: An agent whose controller implements processes that can produce models which may be used for planning, prediction, and goal achievement in the task-environment, may evaluate their usefulness for various purposes through experience. This can be done both in situ, by acting on the world according to what the models prescribe (‘do’) and noting the effect (‘perceive’), as well as – in an agent capable of reflection – per cogitandi through reasoning, where the effects of particular cognitive processes on other cognitive processes is noted (not depicted here). Such an agent may learn to control its attention (‘Focus’), planning and learning processes. The usefulness of its models for these various internal processes, as well as achieving domain-specific goals and sub-goals, is measured in the same way, using the reliability measure of the models (Equation 1).

for what *might be the case*—and that’s why they need to be revisable, augmentable, verifiable, and contextualizable.) To assess a model’s usefulness, however, we are not limited to prediction through deduction; this may also be done through abduction, induction and analogy—ampliative reasoning.²³ We will first look at abduction, then analogy (induction will not be addressed in detail here).

Consider a system where a cause α has two effects, β and γ (figure 4). We assume that to the agent, α appears before β and γ , but β and γ appear together. Four models can be used to describe what the agent experiences every time it observes these variables:

$$m_1[\beta \rightarrow \gamma] \quad m_2[\gamma \rightarrow \beta] \quad m_3[\alpha \rightarrow \beta] \quad m_4[\alpha \rightarrow \gamma]$$

Any of these models will correctly predict observed events: If you observe β you will observe γ , and vice versa; if you observe α you will observe β and γ . They can be combined to cover the full experience with all variables: m_1 and m_3 ; m_2 and m_4 ; m_3 and m_4 . However, not all of them represent the actual causal relations, and not all of them can be used to achieve goals in the α - β - γ domain. Let’s say that α is a light switch, β is a light bulb, and γ is the room appearing when lit up by the light. If you presses α , the light turns on. If you press α again, it turns off. If you want to stop γ from appearing it does not help to manipulate β , or vice versa (you could break the lightbulb, but that goes beyond the agent’s knowledge in this example)—to manipulate either β or γ the only variable that will help achieve the right sub-goal is α , as specified by m_3 and m_4 .

23. Peirce’s use of the concept of ‘ampliative reasoning’ included abduction, induction and analogy (cf. Psillos 2011); ours adds (corrigible) deduction to that list.

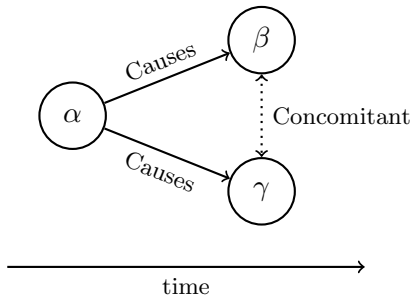


Figure 4: Relations between three variables or patterns α , β and γ (from Thórisson and Talbot 2018b). See text for explanation.

So models that predict correctly²⁴ may still be wrong about the causes of the predictions; the LHS of a model might not be an exclusive cause of the RHS. For the LHS to be an effective²⁵ (necessary and sufficient) cause of the RHS in condition Z , the RHS must disappear in the absence of LHS and appear in the presence of the LHS, given no changes in Z . A good predictor that does strive for (semi-) deterministic necessary-and-sufficient causes is no good for getting things done as it cannot be relied on for effective action (we remind the reader of the New York Mayor who banned summer ice cream sales in Central Park to stop muggings, because the evidence showed strong correlation between the two). To create models that are not only good for prediction but also for getting things done, backward chaining – reading a *CRM* from right to left – tells us how to use a model to make a plan: If the RHS is a goal, or part of a causal chain constituting a potential plan, the model’s LHS is a sub-goal *if and only if it corresponds to an effective cause in the world it models*. This is in effect abduction.

When each of these models, m_1, m_2, m_3 and m_4 is used for *both* prediction and goal achievement, models m_1 and m_2 will be deleted due to their incorrect predictions: It does not help to manipulate either β or γ directly to get γ to appear. What remains is a model $m_x : [fact(light = off) \wedge fact(press(\alpha)) \rightarrow fact(light = on)]$ that tells the agent that “buttons can turn on lights”.²⁶ Repeating such operations will leave only models that capture useful causal relations in the domain, to the extent that this can be represented as relationships between *observable* variables: If the agent wants the room to appear it presses the button using model m_x . For any particular such use case, the model would be instan-

24. In the physical world the “correctness” of models can only be ascertained by their *usefulness*, and we cannot ever be sure that useful models developed from experience are “correct,” no matter how useful. Nevertheless, this way of wording it is useful for simplification.

25. In the limit an ‘effective’ identified cause is deterministic; in the physical world it is sufficient to find factors that are useful, meaning they work most of the time and don’t have a known or predicted potential for undesired side effects.

26. More specifically, the generality of the model, e.g. whether it reads “this button will turn on that light” or “buttons of this kind tend to turn on lights of that kind” will be determined by how the LHS and RHS patterns are represented. There are other aspects which we omit for clarity, including information about time, context, and transformation functions of variable values for both deduction and abduction. Details can be found in Nivel et al. (2015, 2013c).

tiated with the variables bound to the particular situation and its usefulness subsequently recorded, to accurately reflect its reliability.

Once candidate models – really, hypotheses of how the world hangs together – have been created, they might predict incorrectly, in which case they will get an increasingly lower score over time and soon be deleted (they will also be deleted if other models exist that work better). Through the creation of a number of such models, with patterns on each side referencing internal as well as external variables, and testing the models through a process of deduction and abduction, a network of *useful* models is built up. Notice that a temporally-grounded bi-directional model that can be relied on to get things done will *also* be good for prediction—i.e. we do not lose the predictive power of the model set through this pruning, we simply remove those that are no good for planning and guiding action. Such processes are explained further in Thórisson and Talbot (2018b). We conclude that:

§6 *Unified abduction and deduction centered on a single model can isolate causal relations in experience data.*

For a newborn, as well as a learner with prior partial knowledge of a phenomenon or situation, the actions it takes must have a realistic potential to result in changes that are observable to it. What must be learned in this knowledge bootstrapping is information that supports both prediction and planning, because otherwise the learner cannot use learned predictions to affect the world—which, after all, is a fundamental reason for intelligence. In other words, pragmatic knowledge is fundamental to knowledge bootstrapping. Thus, a necessary job of an autonomous general learner is to reverse-engineer causal chains. We further conjecture that this process lies at the heart of all novelty learning—that when faced with highly novel phenomena, in really novel situations, an autonomous general learner will rely on these principles. We summarize this as:

§7 *Extraction of causal relations from experience data is fundamental to knowledge bootstrapping.*

Having created a lot of models, some – possibly most – will be irrelevant to any particular task or subtask at a particular time. For this the controller needs a practical, efficient way to find the most relevant ones at any point in time. This is done through resource control mechanisms (attention), anchored in the present by the current state \mathcal{S}_{now} , and in the future by the agent’s active yet unachieved sub-goals $\langle \mathcal{G}, \mathcal{G}^- \rangle$.²⁷ Similarity of prior experience to the present state is used to determine what knowledge is relevant. In our conceptualization, similarity and relevance are close cousins: Models that share variables with a current situation and active goals are highly likely to be relevant, because they “talk about” how some patterns in the current situation may be transformed over time, and how the controller may achieve transformations that result in goal achievement. Relevance can be computed by a similarity function that is anchored at one end (‘now’) by the current situation \mathcal{S} and at the other end (the future) by the goals of the agent $\langle \mathcal{G}, \mathcal{G}^- \rangle$ that are active (active goals are those that are being actively pursued at any point in time, the most relevant ones being all immediate sub-goals).

27. Through the interplay of attention, planning and learning, the variables of the world that matter to the agent’s knowledge and purpose are targeted, sidestepping the issue of information overload, complete simulation of the world, etc.; *Te* is made tractable by bounding it by the needs of agency, cognition, and perception.

For finding similarity between patterns we define Ψ , a multi-dimensional comparison computation using ampliative reasoning (Sheikhlar et al. 2020) that takes two or more patterns and returns their similarity.²⁸ Running $\Psi(\vartheta, \mathcal{M}, \mathcal{G})$, where ϑ is a target pattern (from a model, a goal, percept, or other source), \mathcal{M} is a set of models and \mathcal{G} a set of goals (goals are also defined by patterns), produces a similarity gradient over them that serves as an indicator of relevance, because only if the models share (subsets of, or entire) patterns may they reference shared phenomena. If we define *aspects* of a phenomenon as a subdivision of it, $\mathcal{A} \subset \Phi$, that is of pragmatic importance to the agent’s goals and tasks, every aspect may be verified through multiple percepts predicted by relevant models. Aspects that an agent can predict this way are highly likely to be familiar to the agent, \mathcal{A}_{fam} ; those it can’t predict are novel, \mathcal{A}_{nov} (Sheikhlar et al. 2020), other things being equal.

In Sheikhlar et al. (2020) we detail a method for computing analogies employing the above principles. With the addition of ways to narrow down the relevant models, patterns, aspects, and percepts, the similarity computations can be used to create analogies at any level of detail. On the basis of further similarity computations, it can proceed to (a) dissect the novel aspects and generate new models, possibly patterned after similar yet incorrect models,²⁹ in an attempt to (b) fill in missing knowledge gaps. To summarize this principle:

§8 *Bootstrapping on variables that are potentially causally related is necessary for all novelty learning.*

Correlation is the most basic indication of causal relation, so as already discussed, this bootstrapping is done by observing correlation and creating initial *CRMs* based on those.

6. Seed-Programmed Autonomous General Learning

As informationally rich open-ended complex environments cannot be axiomatized beforehand, very few assumptions can be made about the knowledge that a system may have to acquire in such an environment: The information available to be perceived and the behavioral challenges that the system’s future environment(s) may present, may be of many types, and the specifics of these types cannot be known beforehand. The generality of the system is thus constrained by its ability to deal with this potentially large set of information, as well as its own skills, in an organized manner: the less specific to particular types of information the knowledge representation is, the greater the system’s potential for being general. All observations are by definition specific; an efficient way to generalize them is through reasoning (not just induction—we have seen that induction must be accompanied by other forms of reasoning to be of any use).

In a world where rule *A* sometimes holds for situations of type *T* and rule *B* for other times, and where *A* and *B* are in some way incompatible, this may mean that something

28. This comparison function is implemented through several mechanisms in our cognitive architecture; its description is simplified here for clarity. In addition to these methods, the controller needs to implement resource management mechanisms (‘attention’ or ‘focus’) to efficiently and effectively manage large sets of models (Nivel and Thórisson 2013c, Helgason et al. 2012); this forms an integral part of our theory but will not be addressed here.

29. Or using other methods; an agent whose knowledge is reasonably consistent and comprehensive, the need to resort to random generation will seldom if ever be preferable to one that is in some way guided by the current knowledge.

is amiss—either the models targeting the rules are in some way incorrectly constructed, the observations on which they are based are faulty, the models are missing some facts, or a third non-existent model is needed to qualify the use of the models based on context, as could happen when there is a context change that has gone undetected (for lack of an appropriate third model). A search for a variable or pattern that allows predicting when either A or B should be used may lead to more effective models, the creation of brand new ones, or modification of partially faulty or incomplete ones. In every case this involves a search for ‘indicative evidence’ of variables or patterns that correlate (negatively or positively) with the cases where A worked well and B not (and vice versa), but that have so far been ignored. (The search can be implemented by spreading activation that makes salient patterns and variables that share features (values, value ranges, variables) with the conditions under which (either) A or B are relevant, sifting through these for evidence that might separate them.)³⁰ A world with hierarchical rules allows systematic generation and testing of hypotheses for these using logic, even if the rules must be inferred from observations of changes in variables over time.

To rely on logic means doing reasoning. In our theory reasoning permeates the knowledge creation and management processes from a very low level of detail and upwards, using all available forms of (ampliative)³¹ reasoning—deduction, abduction, induction and analogies. To emphasize the fine-grain level of detail, and the contrast with higher level of (human-like) conscious reasoning, we call them *micro-ampliative* reasoning methods.³² Micro-ampliative reasoning supervenes on the fine-grain causal-relational information discussed above, ensuring consistency at a very fine-grain level, based on similarity (proximity in analogy space).

In short, the purpose and principles of ampliative reasoning methods in our approach may be listed as follows:

- Deduction, to produce plausible turn of events: *Simulation of causal chains from a given state through forward chaining.*
- Abduction, to find plans that work: *Constraint-based backward-chaining from a given end goal through plausible causal chains that can lead to it from a given starting state.*
- Induction, to generalize observations: *Preferring general rules, then reducing scope in light of new evidence.*

30. If the *CRMs* are small, and the hidden variables not extremely ‘buried’ in the data available to the agent, this will typically not be taxing on the system’s processing and can be achieved while learning. This is one of the reasons there is a ‘sweet spot’ window in which animals learn: The search for the relevant differentiators – the differences that make a difference – is not too taxing in light of existing knowledge. Teachers can help learners reduce such search significantly through systematic candidate elimination. The candidate correlating-and-differentiating factors are subsequently used to create new models that attempt to capture a causal relation between observable evidence (yet so far ignored) for when to use either rule A or B .

31. Peirce’s suggested ‘ampliative reasoning’ to include abduction, induction and analogy making (Psillos 2011); we include (corrigible) deduction as well.

32. It is not a coincidence that recent software development methods are leaning towards “microservices” with fine-grain semantics (Zhelev and Rozeva 2019); the argument for that rests on this same foundation.

- Analogy, for discretionary comparisons of knowledge and percepts: *Identifying relevant knowledge through similarity computations and handling novelty.*
- Unified knowledge management: *Combined backward-forward chaining to hone in on causal relations.*
- Attention-driven similarity computations: *Bounding ampliative (micro-)reasoning at “two ends,” the perceived here-and-now and unachieved, active goals.*

The kind of reasoning we are talking about is a kind of “micro-reasoning”: Fine-grain (‘pee-wee’) causal-relational models are continuously organized via these processes, and therefore no major (dangerous or practically prohibitive) overhaul of the knowledge is needed, implementing a bounded recursive self-improvement scheme (Thórisson and Nivel 2009).

To improve incomplete and incorrect knowledge with increased experience – accumulated evidence – knowledge acquisition processes in this scheme bring already-acquired knowledge opportunistically (but systematically) and constantly towards greater coherence and consistency (implementing a kind of “antifragile” system (Taleb 2012)). For the initial creation of knowledge, and its subsequent usage, expansion, modification, unification, and deletion, construction mechanisms are self-guided, capable of self-supervised “surgical” operation on existing knowledge. To allow appropriate application to any situation, the knowledge must be associated with observed variables and patterns in the environment, which means the system must be able to selectively compare knowledge structures. Flexible and selective comparison imply in effect the support of analogies. If the learner is to get better at making analogies this means that analogy making mechanisms themselves must also be comparable: “For situation X and purposes Y , analogies of type A work better than of type B ”. This means that the learner must be capable of *introspection*.

An implementation of this general scheme was used in our Autocatalytic Endogenous Reflective Architecture (AERA; Nivel et al. 2013c, Nivel et al. 2013b, Nivel and Thórisson 2013b), in a way that allows self-modeling: Since *CRMs* can reference each other, via their LHS and RHS patterns, any plan the system creates can involve the internal planning mechanisms (e.g. which models are used during the plan) as well as how to act with respect to the environment—while the system keeps track of the referents of each of the models, the mechanism itself does not care what the references are, in essence creating a “syntax for thinking” (Dindo et al. 2013). During a normal run of AERA, a multitude of models are being used simultaneously to continuously predict and plan, in parallel.

In one evaluation setup of AERA, our S0 agent was given the task to learn how to meet the demands of a human telling it to move one of four objects around on a (virtual) tabletop (see Figure 5). S0 received a stream of data generated from the computer graphics and speech. The speech included a few phrases (e.g. “Put that [deictic gesture 1] there [deictic gesture 2],” “Put the red sphere here,” “Take the blue cube ... and put it next to the red sphere [deictic gesture]”) and fairly simple grammar.³³ There were four objects,

33. The whole grammar that the humans followed (but was not provided to S0 and thus had to be learned):

$utterance = [PRT_1^*, PRT_2, TKS]; PRT_1 = [take, DET, CLR^*, SHP, and];$ and
 $PRT_2 = [put, [it | CLR^*, SHP], POS],$ where $DET = [that | a | the], CLR = [blue | red],$
 $SHP = [sphere | cube], POS = [[nextto, [DET, CLR^*, SHP]] | [[here | there], DEICT]];$ where
 $DEICT$ is a co-temporal pointing gesture, TKS are the words “Thank you,” and star means optional.

two cubes, blue and red, and two spheres, also blue and red. The seed of S0 contained one simple drive (to please the human, operationalized as the goal of receiving the pattern “thank you”), a small ontology telling it the names of the objects in the graphics database, and 6 CRMs. In this case, S0’s drive spans only one turntaking cycle, so it was set to re-inject (become re-activated) every time it was achieved, allowing the dialog to go on for multiple turns. The seed contained no information about grammar or syntax, neither the form or function of deictic (pointing) gestures, the meaning or interpretation of ellipsis, nor what words referring to placement meant—all of these had to be inferred and learned from observation by S0, given its seed. After observing two humans playing a game of “put that there” for about 2.5 minutes, S0 had created 20 models of its own, for a total of 26 *CRMs* that allowed it to perform the task flawlessly in interaction with a human (and more efficiently to our surprise, as it turned out, because it did not lift objects off the virtual table, but rather slid them around—something it had never observed being done by the humans but rather discovered while creating the models for achieving the goals of the task).

In another evaluation scenario, our AERA-S1 agent was given a seed to learn to conduct a TV-style interview. The interview was conducted between the agent and a human participant in cyberspace—akin to a two-person video conference. S1 learned by watching a human interviewer and interviewee, the latter of who was knowledgeable about recycling. No information was given to S1 about syntax, turntaking, deictic gestures, anaphora, or how to construct an answers to questions; the seed contained only five drives, 26 models, and a small ontology of graphical elements. After 20 hours observation, S1 could perform the interview with a human in realtime, in either role of interviewer or interviewee, flawlessly (Nivel et al. 2014c).³⁴ It had auto-created 1400 CRMs that constituted its own rules for syntax that completely covered the language use it had observed, including some generalizations, so that it knew how to direct the interviewee’s attention to objects using three different deictic methods (picking up objects; pointing to them with an index finger; waving a hand in their direction), as well as use such information to infer which objects are referenced by the other interlocutor when using “it,” “that,” “this,” “here” and “there.” It could also handle continuity, as when the interviewer said “tell me more,” using metamodels (sets of *CRMs*) to keep track of context. Because no grammar was provided up front, generation of full sentences, in response to e.g. a question, included backward chaining of several *CRMs* that predicted which words should go where, in light of the super-goal of answering the question (this goal itself not existing in the seed in the beginning and thus being learned by the system from observation), as well as their exact timing (to a level sufficient for fluid dialog). The vastly larger number of models needed by S1, in comparison to S0, is partly due to a much larger number of words, but more so due to the complexity of representing syntax of long sentences correctly: There is a lot of sentence structures and variations of saying the same – or similar – things, and this must be represented by a relatively large number of models because their LHS and RHS typically, by design,³⁵ do not contain more than a few variables each.

34. Detailed descriptions of the architecture can be found in Nivel et al. (2015), Nivel et al. (2014b), Nivel et al. (2013b), Nivel and Thórisson (2013b) and Thórisson and Talbot (2018a); results of experiments with S1 are detailed in Nivel et al. (2014c) and Nivel et al. (2014d).

35. This is done both because of the ‘peewee surgical precision’ and semantic modularity that we desire in updating the knowledge, as well as requirements of runtime efficiency and interruptability.

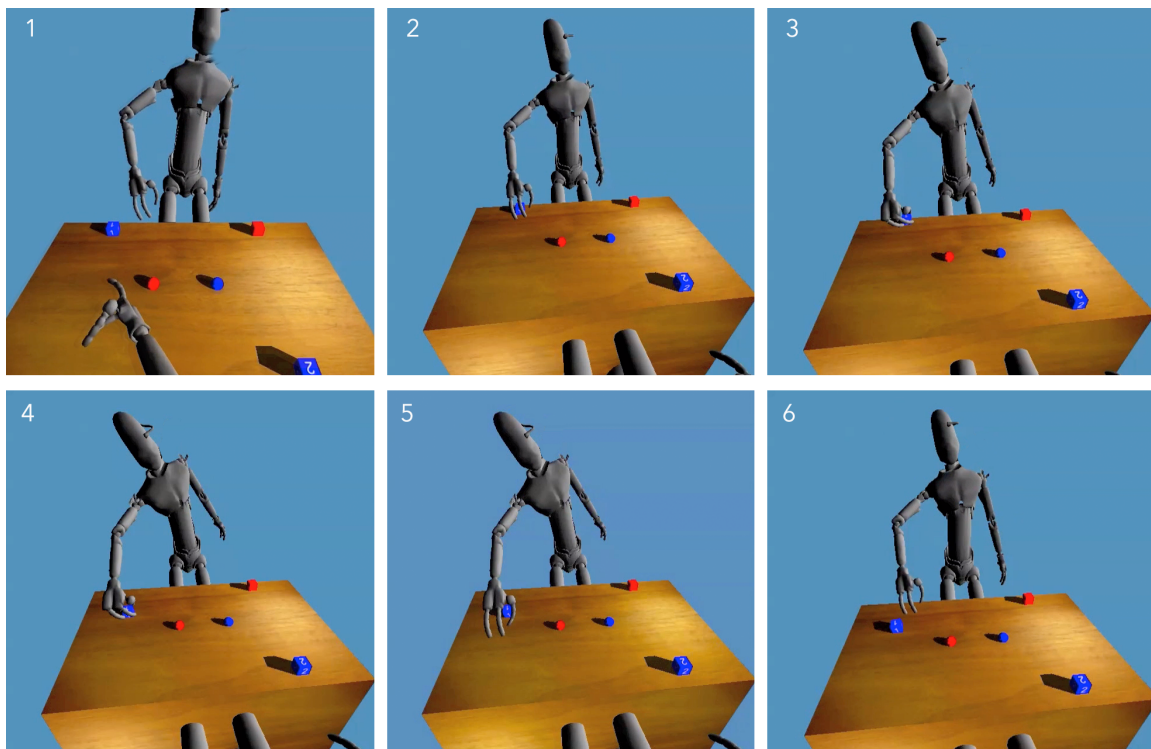


Figure 5: A 10-second snippet from a realtime interaction between a human and S0, seen from the human’s side (human movements were tracked to control an avatar in realtime; voice signal processed with a speech recognizer and a prosody tracker). In frame 1 the human tells S0 to “Put a blue cube there [deictic gesture]” in frame 1, pointing to a position on the virtual table with the index finger, after which S0 resolves the two deictic references (there are two blue cubes, and one new location), grabs the correct cube and moves it to the indicated location. The knowledge for achieving its defined top-level goal of “pleasing the interviewer” was acquired by S0 by watching humans do the same over a period of a couple of minutes. The timing of each of the six frames is (in seconds, starting with the end of the utterance in frame 1): 0, 5, 6, 7, 8, 10. S0 held the position in frame 5 until hearing the words “Thank you,” uttered at the 9th second. Unlike the humans it observed, S0 learned through induction that objects did not need to be lifted to be moved. This demonstration ran on a four-core 2010 Intel laptop.

7. Conclusions

Our aim is to create general machine intelligence (GMI). To get there requires meeting a number of requirements, some of which are generally agreed upon in the research community (cf. Thórisson et al. 2015, Laird and Wray III 2010), others which undoubtedly are less obvious and therefore more controversial. Due to the requirement of such GMI having to eventually address the physical world, we have argued for a non-axiomatic approach to autonomous cumulative learning and presented ideas for how this may be achieved. A major thrust of the arguments in favor of our approach comes from the need for semantic closure and semantic compositionality: Representations that carry operational semantics down to fine-grain ‘peewee’ levels, allowing precision modification and updating of arbitrary subsets of knowledge required for implementing efficient and effective autonomous cumulative learning. The need for cumulative learning from the fact that in the physical world information presentation is always incremental (“life-long”) and large updates of the knowledge, when new (and sometimes contradicting information) comes to light, will be too expensive to be practically implementable. Our proposed approach enables autonomous bootstrapping from existing knowledge, allowing acquisition of novel information. The frameworks presnted in AERA (Nivel et al. 2013c) and NARS (Hammer and Lofthouse 2020, Wang 2006), share many of the fundamental principles discussed here. Both have been demonstrated to handle underspecified tasks and task-environments, and unknown transform functions only accessible over time via multi-dimensional observable variables. While somewhat different in their implementation, results from this work presents evidence that this general approach is both viable and scalable. The ability of these systems to use contexts not specified beforehand to help with their learning of novel spatio-temporal tasks further lends credibility to the approach. Future work involves further development of these principles, especially w.r.t. flexible analogy making, cumulative learning, concept creation and higher-level communication.

Acknowledgments

I am deeply indebted to my friend and collaborator Eric Nivel for countless and extended in-depth discussions about these topics and related ideas over the past fourteen years. I would like to thank Bas Steunebrink, Jeff Thompson, Finnur Dellsén and Pei Wang, as well as my students Jordi Bieger, Arash Sheikhlari, Leonard Eberding, Helgi P. Helgason, Throstur Thorarensen, and Hjörleifur Henriksson for valuable discussions about many of these subjects. Thanks also to the members of the HUMANOBS consortium for their deep insights (you know who you are!). Thanks to Leonard Eberding, Henry Minsky, Olivier Georgeon and Brendan Ruff for valuable feedback on drafts of this paper. This work was supported in part by the Department of Computer Science at Reykjavik University, the Icelandic Institute for Intelligent Machines, and Cisco Systems Inc., as well as prior grants too numerous to recount. —*KRTh*

References

Aristotle. *De Anima*. 350 BCE.

- W. Ross Ashby. *An introduction to cybernetics*. Chapman Hall, 1956.
- W. Ross Ashby. Requisite variety and its implications for the control of complex systems. *Cybernetica*, 1:83–99, 1958.
- Dmitry Balakhonov and Jonas Rose. Crows rival monkeys in cognitive capacity. *Scientific Reports*, 7:2045–2322, 2017. doi: 10.1038/s41598-017-09400-0.
- T. F. Beernaert, A. E. Bayrak, L. F.P. Etman, and P. Y. Papalambros. Framing the concept of autonomy in system design. *Proceedings of International Design Conference, DESIGN*, 6:2821–2832, 2018. ISSN 18479073. doi: 10.21278/idc.2018.0281.
- Tarek R. Besold and Ute Schmid. Why generality is key to human-level artificial intelligence. *Advances in Cognitive Systems*, 4:13–24, 2016.
- Jordi Bieger and Kristinn R. Thórisson. Evaluating understanding. In *IJCAI Workshop on Evaluating General-Purpose AI, Melbourne, Australia*, 2017.
- Jordi Bieger, Kristinn R. Thórisson, and B. Steunebrink. Evaluating understanding. In *IJCAI Workshop on Evaluating General-Purpose AI, Melbourne, Australia (2017 to be accepted)*, 2017.
- Jordi E. Bieger and Kristinn R. Thórisson. Task analysis for teaching cumulative learners. In *Proc. Artificial General Intelligence*, pages 21–31. Springer International Publishing, 2018.
- Giovanni Pezzulo Cristiano Castellfranchi. Thinking as the control of imagination: A conceptual framework for goal-directed systems. *Psychological Research*, 73:559–577, 2009. doi: 10.1007/s00426-009-0237-z123.
- Balakrishnan Chandrasekaran, John R. Josephson, and V. Richard Benjamins. *Ontology of tasks and methods*. 1999.
- Finnur Dellsén. Scientific progress: Four accounts. *Philosophy Compass*, 13, 2018.
- Haris Dindo, Giuseppe La Tona, Eric Nivel, Giovanni Pezzulo, Antonio Chella, and Kristinn R. Thórisson. Simulation and anticipation as tools for coordinating with the future. *Advances in Intelligent Systems and Computing*, 196 AISC:117–125, 2013. ISSN 21945357. doi: 10.1007/978-3-642-34274-5_24.
- Werner Dubitzky, Olaf Wolkenhauer, Kwang-Hyun Cho, and Hiroki Yokota, editors. *Encyclopedia of Systems Biology*. Springer Publishers, 2013.
- Leonard M Eberding, Arash Sheikhlari, Kristinn R Thórisson, and Sindri P Andrason. Sage: Task-environment platform for evaluating a broad range of AI learners. In *International Conference on Artificial General Intelligence*. Springer, 2020.
- David Garlan, Robert Allen, and John Ockerbloom. Architectural mismatch or why it’s hard to build systems out of existing parts. In *Software Engineering, 1995. ICSE 1995. 17th International Conference on*, pages 179–179, 1995.

- Olivier L Georgeon and Alexander Riegler. ScienceDirect CASH only: Constitutive autonomy through motorsensory self-programming. *Cognitive Systems Research*, 58:366–374, 2019. ISSN 1389-0417. doi: 10.1016/j.cogsys.2019.08.006. URL <https://doi.org/10.1016/j.cogsys.2019.08.006>.
- James J. Gibson. *The senses considered as perceptual systems*. SHoughton Mifflin, 1966.
- Ben Goertzel. Toward a formal characterization of real-world general intelligence. In *Proceedings of the 3rd Conference on Artificial General Intelligence, AGI*, pages 19–24, 2010.
- Ben Goertzel. Artificial general intelligence. In *Scholarpedia*, volume 10, page 31847. 2015.
- Patrick Hammer and Tony Lofthouse. ‘opennars for applications’: Architecture and control. In *Proceedings of the 13th International Conference on Artificial General Intelligence*, 2020.
- Helgi Páll Helgason and Kristinn R. Thórisson. Attention Capabilities for AI Systems. In *ICINCO (1)*, pages 281–286, 2012.
- Helgi Páll Helgason, Eric Nivel, and Kristinn R. Thórisson. On attention mechanisms for AGI architectures: a design proposal. In *Artificial General Intelligence*, pages 89–98. Springer, 2012.
- Helgi Páll Helgason, Kristinn R. Thórisson, Eric Nivel, and Pei Wang. Predictive Heuristics for Decision-Making in Real-World Environments. In *Proceedings of the Sixth Conference on Artificial General Intelligence*, volume 7999 of *Lecture Notes in Computer Science*, Beijing, China, 2013. Springer.
- Francis Heylighen and Cliff Joslyn. Cybernetics and second order cybernetics. *Encyclopedia of physical science & technology*, 4:155–170, 2001.
- John E. Laird and Robert E. Wray III. Cognitive Architecture Requirements for Achieving AGI. pages 1–6, 2010. doi: 10.2991/agi.2010.2.
- John E. Laird and Robert E. Wray III. Cognitive architecture requirements for achieving AGI. In *Proc. of the Third Conference on Artificial General Intelligence*, pages 79–84, 2010.
- W.F. Lawless, R. Mittu, D. Sofge, and S. Russell. *Autonomy and Artificial Intelligence: A Threat or Savior?* Springer, 2017.
- E. E. Lee. Computing needs time. *Commun. ACM*, 52:70–79, 2009.
- Juan Carlos Letelier, María Luz Cárdenas, and Athel Cornish-Bowden. From L’Homme Machine to metabolic closure: Steps towards understanding life. *Journal of Theoretical Biology*, 286(1):100–113, 2011. ISSN 00225193. doi: 10.1016/j.jtbi.2011.06.033.
- Allen Newell. *Unified theories of cognition*. Harvard University Press, Cambridge, MA, 1994.

- E. Nivel, K. R. Thórisson, and et al. Autonomous acquisition of natural language. *IADIS International Conference on Intelligent Systems & Agents*, pages 58–66, 2014a.
- Eric Nivel and Kristinn R. Thórisson. Self-Programming: Operationalizing Autonomy. In *Proceedings of the 2nd Conf. on Artificial General Intelligence*, 2008.
- Eric Nivel and Kristinn R Thórisson. Towards a programming paradigm for control systems with high levels of existential autonomy. In *International Conference on Artificial General Intelligence*, pages 78–87. Springer, 2013a.
- Eric Nivel and Kristinn R. Thórisson. Replicode: A constructivist programming paradigm and language. Technical RUTR-SCS13001, Reykjavik University School of Computer Science, 2013b.
- Eric Nivel and Kristinn R. Thórisson. Towards a Programming Paradigm for Control Systems with High Levels of Existential Autonomy. In Kai-Uwe Kühnberger, Sebastian Rudolph, and Pei Wang, editors, *Proceedings of the Sixth Conference on Artificial General Intelligence*, volume 7999 of *Lecture Notes in Artificial Intelligence*. Springer, Beijing, China, 2013c.
- Eric Nivel, Kristinn R. Thórisson, H. Dindo, G. Pezzulo, M. Rodriguez, C. Corbato, B. Steunebrink, D. Ognibene, A. Chella, Jürgen Schmidhuber, Ricardo Sanz, and Helgi Páll Helgason. Autocatalytic Endogenous Reflective Architecture. Technical RUTR-SCS13002, Reykjavik University School of Computer Science, Reykjavik, Iceland, 2013a.
- Eric Nivel, Kristinn R Thórisson, Bas Steunebrink, Harris Dindo, Giovanni Pezzulo, Manuel Rodriguez, Carlos Corbato-Hernandez, Dimitri Ognibene, Jörgen Schmidhuber, Ricardo Sanz, Helgi P. Helgason, and Antonio Chella. Bounded recursive self-improvement. *Tech report RUTR-SCS13006, Reykjavik University – School of Computer Science*, 2013b.
- Eric Nivel, Kristinn R Thórisson, Bas Steunebrink, Harris Dindo, Giovanni Pezzulo, Manuel Rodriguez, Carlos Corbato-Hernandez, Dimitri Ognibene, Jörgen Schmidhuber, Ricardo Sanz, Helgi P. Helgason, and Antonio Chella. Autocatalytic endogenous reflective architecture. *Tech report RUTR-SCS13002, Reykjavik University – School of Computer Science*, 2013c.
- Eric Nivel, Kristinn R Thórisson, Bas Steunebrink, Harris Dindo, Giovanni Pezzulo, Manuel Rodriguez, Carlos Corbato-Hernandez, Dimitri Ognibene, Jörgen Schmidhuber, Ricardo Sanz, Helgi P. Helgason, and Antonio Chella. Bounded seed-AGI. pages 85–96, 2014b.
- Eric Nivel, Kristinn R Thórisson, Bas Steunebrink, Harris Dindo, Giovanni Pezzulo, Manuel Rodriguez, Carlos Corbato-Hernandez, Dimitri Ognibene, Jörgen Schmidhuber, Ricardo Sanz, Helgi P. Helgason, Antonio Chella, and Gudberg K. Jonsson. Autommous acquisition of natural language. In *Proceedings of the IADIS International Conference on Intelligent Systems & Agents*, pages 58–66, 2014c.
- Eric Nivel, Kristinn R. Thórisson, Bas R. Steunebrink, Haris Dindo, Giovanni Pezzulo, Manuel Rodríguez, Carlos Hernández, Dimitri Ognibene, Jürgen Schmidhuber, Ricardo

- Sanz, and others. Bounded Seed-AGI. In *Artificial General Intelligence*, pages 85–96. Springer, 2014d.
- Eric Nivel, Kristinn R. Thórisson, Bas Steunebrink, and Jürgen Schmidhuber. Anytime Bounded Rationality. In Jordi Bieger, Ben Goertzel, and Alexey Potapov, editors, *Proceedings of The Eighth Conference on Artificial General Intelligence*, volume 9205 of *Lecture Notes in Artificial Intelligence*, pages 121–130. Springer, Berlin, Germany, 2015.
- F.G.P. Patterson and W. Gordon. *Twenty-Seven Years of Project Koko and Michael*, pages 165–176. Springer US, Boston, MA, 2001. ISBN 978-0-306-47461-3. doi: 10.1007/0-306-47461-1_15. URL https://doi.org/10.1007/0-306-47461-1_15.
- I. P. Pavlov. *The work of the digestive glands*. Griffin, 1902.
- J. Pearl. *do*-calculus revisited. In Nando de Freitas and Kevin Murphy, editors, *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence*, pages 4–11, Corvallis, OR, 2012. AUAI Press.
- Judea Pearl. Bayesianism and causality, or, why I am only a half-Bayesian. volume 12, pages 19–36, Corvallis, OR, 2001. Kluwer Academic Publishers.
- Judea Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, New York, NY, USA, 2nd edition, 2009. ISBN 052189560X, 9780521895606.
- Judea Pearl, editor. *The Book of Why: The New Science of Cause and Effect*. Penguin Books Ltd, 2019.
- Jean Piaget. *The origin of intelligence in the child*. Routledge & Paul, 1953.
- Jean Piaget. Piaget’s Theory. In Bärbel Inhelder, Harold H. Chipman, and Charles Zwingmann, editors, *Piaget and His School*, Springer Study Edition, pages 11–23. Springer Berlin Heidelberg, 1976. ISBN 978-3-540-07248-5 978-3-642-46323-5.
- John L. Pollock. Defeasible reasoning and degrees of justification. *Argument and Computation*, 1(1):7–22, 2010.
- Stathis Psillos. An explorer upon untrodden ground: Peirce on abduction. In *Handbook of the History of Logic*, volume 10, pages 117–151. Elsevier, 2011.
- H. Putnam. *Reason, Truth and History*. Cambridge University Press, 1981.
- Darryn J. Reid, M. J. Pilling, and Vladimir Ivancevic. *Mathematics Of Autonomy: Mathematical Methods For Cyber-physical-cognitive Systems*. World Scientific, 2018.
- Ken Richardson, editor. *Models of Cognitive Development*. Psychology Press, 1998.
- Luis Mateus Rocha. Syntactic Autonomy: Why There is No Autonomy Without Symbols and How Self-Organizing Systems Might Evolve Them. *Annals of the New York Academy of Sciences*, 901(1):207–223, 2010. doi: 10.1111/j.1749-6632.2000.tb06280.x.

- W. Ross Ashby Roger C. Conant. Every good regulator of a system must be a model of that system. *Int. J. Systems Sci.*, 1(2):89–97, 1970.
- Arash Sheikhlari, Kristinn R Thórisson, and Leonard M Eberding. Autonomous cumulative transfer learning. In *International Conference on Artificial General Intelligence*. Springer, 2020.
- Linda Smith and Michael Gasser. The development of embodied cognition: Six lessons from babies. *Artificial life*, 11(1-2):13–29, 2005.
- Bas R. Steunebrink, Kristinn R. Thórisson, and Jürgen Schmidhuber. Growing recursive self-improvers. In *Proceedings of Artificial General Intelligence*, pages 129–139, 2016.
- Nassim Nicholas Taleb, editor. *Antifragile: Things That Gain from Disorder*. Random House, 2012.
- A. Taski. The semantic conception of truth. *Philosophy and Phenomenological Research*, 4: 341–375, 1944.
- Kristinn R. Thórisson. Modeling multimodal communication as a complex system. In *Modeling Communication with Robots and Virtual Humans*, pages 143–168. Springer, 2008.
- Kristinn R. Thórisson. A New Constructivist AI: From Manual Methods to Self-Constructive Systems. In Pei Wang and Ben Goertzel, editors, *Theoretical Foundations of Artificial General Intelligence*, volume 4 of *Atlantis Thinking Machines*, pages 145–171. Atlantis Press, Amsterdam, The Netherlands, 2012.
- Kristinn R. Thórisson. Machines with Autonomy & General Intelligence: Which Methodology? In *Proceedings of the Workshop on Architectures for Generality and Autonomy 2017*, Melbourne, Australia, August 2017.
- Kristinn R Thórisson. Discretionarily constrained adaptation under insufficient knowledge & resources. *Journal of Artificial General Intelligence*, 11:7–13, 2020. doi: 10.2478/jagi-2020-0003.
- Kristinn R. Thórisson and Helgi P. Helgason. Cognitive architectures & autonomy: A comparative review. *Journal of Artificial General Intelligence*, 3:2:1–30, 2012.
- Kristinn R. Thórisson and Eric Nivel. Achieving artificial general intelligence through peewee granularity. In *Proc. of the 2nd Conf. on Artificial General Intelligence*, pages 220–221, 2009.
- Kristinn R. Thórisson and Arthur Talbot. Cumulative learning with causal-relational models. In *Artificial General Intelligence*, pages 227–237, Cham, 2018a. Springer International Publishing.
- Kristinn R Thórisson and Arthur Talbot. Cumulative learning with causal-relational models. In *International Conference on Artificial General Intelligence*, pages 227–237. Springer, 2018b.

- Kristinn R Thórisson, Jordi Bieger, Stephan Schiffel, and Deon Garrett. Towards Flexible Task Environments for Comprehensive Evaluation of Artificial Intelligent Systems & Automatic Learners. In *Proceedings of the International Conference on Artificial General Intelligence*, pages 187–196, 2015.
- Kristinn R. Thórisson, Jordi Bieger, Throstur Thorarensen, Jona Sigrídur Sigurdardóttir, and Bas R. Steunebrink. Why artificial intelligence needs a task theory – and what it might look like. In *Proceedings of the 9th International Conference on Artificial General Intelligence*, pages 118–128, 2016a.
- Kristinn R. Thórisson, David Kremelberg, Bas R. Steunebrink, and Eric Nivel. About understanding. In *Proceedings of the International Conference on Artificial General Intelligence*, pages 106–117, New York, NY, USA, 2016b. Springer-Verlag.
- Kristinn R Thórisson, Jordi Bieger, Xiang Li, and Pei Wang. Cumulative learning. In *Proceedings of the 12th International Conference on Artificial General Intelligence*, pages 198–208, Shenzhen, China, 2019. Springer.
- Jane X. Wang, Zeb Kurth-Nelson, Dhruva Tirumala, Hubert Soyer, Joel Z. Leibo, Remi Munos, Charles Blundell, Dhharshan Kumaran, and Matt Botvinick. Learning to reinforcement learn. *arXiv:1611.05763 [cs, stat]*, November 2016. arXiv: 1611.05763.
- P. Wang. On defining artificial intelligence—author’s response to commentaries. *Journal of Artificial General Intelligence*, 11(2):73–86, 2020. doi: 10.2478/jagi-2020-0003.
- Pei Wang. *Rigid flexibility: The logic of intelligence*. Springer Science, vol. 34, 2006.
- Pei Wang. From NARS to a thinking machine. *Advances in Artificial General Intelligence: Concepts, Architectures and Algorithms*, 157:75–93, 2007.
- Pei Wang. Insufficient Knowledge and Resources-A Biological Constraint and Its Functional Implications. In *AAAI Fall Symposium: Biologically Inspired Cognitive Architectures*. Citeseer, 2009.
- Pei Wang. *Non-Axiomatic Logic: A Model of Intelligent Reasoning*. World Scientific Publishing, Singapore, 2013. ISBN 978-981-4440-28-8.
- Juyang Weng. Developmental robotics: Theory and experiments. *International Journal of Humanoid Robotics*, 1(02):199–236, 2004.
- Jason Winning and William Bechtel. Being Emergence vs. Pattern Emergence: Complexity, control and goal-directedness in biological systems. In S. C. Gibb, R. F. Hendry, and T. Lancaster, editors, *Routledge Handbook of Emergence*, pages 134–144. Routledge, 2019.
- D. M. Wolpert and M. Kawato. Multiple paired forward and inverse models for motor control. *Neural Networks*, 11(7-8):1317–1329, 1998. ISSN 08936080. doi: 10.1016/S0893-6080(98)00066-5.
- Daniel M Wolpert, Zoubin Ghahramani, Michael I Jordan, New Series, and No Sep. An Internal Model for Sensorimotor Integration An Internal Model for Sensorimotor Integration. *Science*, 269(5232):1880–1882, 2007. ISSN 14372096. doi: 10.1055/s-0037-1609201.

Daniel M Wolpert, R. Chris Miall, and Mitsuo Kawato. Internal Models in the Cerebellum. *Transportation*, 44(4):753–772, 2017. ISSN 15729435. doi: 10.1007/s11116-016-9675-9.

Michael Wooldridge and Nicholas R. Jennings. Agent theories, architectures, and languages: A survey. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 890:1–39, 1995. ISSN 16113349. doi: 10.1007/3-540-58855-8_1.

Svetoslav Zhelev and Anna Rozeva. Using microservices and event driven architecture for big data stream processing. In *AIP Conference Proceedings 2172, 090010 (2019)*, pages 090010–1–090010–8, 2019. doi: 10.1063/1.5133587.