Context In Cognitive Hierarchies

Bernhard Hengst
University of NSW

Maurice Pagnucco
University of NSW

David Rajaratnam
University of NSW

Claude Sammut
University of NSW

Michael Thielscher
University of NSW

Abstract

This paper formalises the notion of context and its influence in a cognitive hierarchy. Cognition does not only depend on bottom-up sensor feature abstraction, but also relies on contextual information being passed top-down. Context is higher level information that helps to predict belief state at lower levels. We show how a cognitive hierarchy can model Pearl's belief propagation in causal trees, and demonstrate contextual influence in a novel approach to visually tracking rigid objects.

1 Introduction

There is strong evidence that scaling intelligence necessarily involves hierarchical structures [Ashby, 1952; Brooks, 1986; Dietterich, 2000; Albus and Meystel, 2001; Beer, 1966; Turchin, 1977; Hubel and Wiesel, 1979; Minsky, 1986; Drescher, 1991; Dayan and Hinton, 1992; Kaelbling, 1993; Nilsson, 2001; Konidaris *et al.*, 2011; Jong, 2010; Marthi *et al.*, 2006; Bakker and Schmidhuber, 2004]. A recent approach [Clark *et al.*, 2016] has addressed the formalisation of cognitive hierarchies that allow for the integration of disparate representations, including symbolic and sub-symbolic representations, in a framework for cognitive robotics. Sensory information processing is upward-feeding, progressively abstracting more complex state features, while behaviours are downward-feeding progressively becoming more concrete, ultimately controlling robot actuators.

However, neuroscience suggests that the brain is also subject to top-down cognitive influences for attention, expectation and perception [Gilbert and Li, 2013]. Higher level signals carry important information to facilitate scene interpretation. For example, the recognition of the Dalmatian, and the disambiguation of the symbol /-\ in Figure 1 intuitively show that higher level context is necessary to correctly interpret the images[1]. The human brain is able to make sense of dynamic 3D scenes from light falling on our 2D retina in varying lighting conditions. Replicating this ability is still a challenge in artificial intelligence and computer vision, particularly when objects move relative to each other, occlude each other, and are without texture. Prior, more abstract contextual knowledge is important to help segment images into objects or to confirm the presence of an object from faint or partial edges in an image.

Figure 1: The Dalmation in the top image would probably be indiscernible without being told what to look for. The ambiguous symbol /-\ on the bottom can be interpreted as either an "H" or an "A" depending on the word context.

In this paper we extend the cognitive architecture formalisation in [Clark *et al.*, 2016] by introducing perceptual context that modifies the beliefs of a child node given the beliefs of parent nodes. As an example of the operation of context we prove that Pearl's [1988] belief propagation in causal trees can be embedded into our framework. As another example we demonstrate the use of context in a computer vision task that involves tracking the pose of multiple occluded featureless objects with a 2D camera.

The contributions of this paper are summarised as follows:

1. *The formalisation of the notion of top-down influence in a general cognitive hierarchy. We call the higher level signals* context.
2. *The representation of belief propagation in causal trees [Pearl, 1988] as a cognitive hierarchy with contextual*

[1] Both of these examples appear in [Johnson, 2010] but are also well-known in the cognitive psychology literature.

information.

3. *Instantiation of a cognitive hierarchy for the perception and tracking of objects using context from the system's "mental imagery" modelled by a 3D physics simulator.*
4. *The implementation of the above system using a Baxter robot to track a scene of multiple, possibly occluded featureless objects with its inbuilt 2D arm camera.*

In the rest of this paper we extend the formalisation of the cognitive hierarchy with contextual functions as foreshadowed [Hengst *et al.*, 2016]. The existing formal meta-model of cognitive hierarchies [Clark *et al.*, 2016] does not include a notion of context. To illustrate the functioning of context we show how causal networks can be interpreted as cognitive hierarchies, and describe the use of context in a challenging vision task, tracking the pose of multiple objects.

2 The Architectural Framework

For the sake of brevity the following presentation both summarises and extends the formalisation of cognitive hierarchies as introduced in [Clark *et al.*, 2016]. We shall, however, highlight how our contribution differs from the original. The essence of this framework is to adopt a meta-theoretic approach, formalising the interaction between abstract cognitive nodes, while making no commitments about the representation and reasoning mechanism within individual nodes.

Being a meta-theory agnostic to specific instantiations of modelling and behaviour mechanisms, detailed complexity and scalability analysis is not possible. Nevertheless, at the meta-level two observations can be made. The formal introduction of context in the cognitive hierarchy only adds a context argument to the prediction update but preserves the well defined update process. Secondly, the decomposition of the agent's complete world model and behaviour into a hierarchy of nodes presents a significant reduction in complexity. While we have not addressed how the decomposition can be achieved other than by design, we have demonstrated that an arbitrary cognitive hierarchy can be composed into just two nodes, one being the environment, along with a considerable increase in complexity [Rajaratnam *et al.*, 2016].

2.1 Motivating Example

As an explanatory aid to formalising the use of context in a hierarchy we will use the disambiguation of the symbol /-\ in Figure 1 as a simple running example. This system can be modelled as a two layer causal tree updated according Pearl's Bayesian belief propagation rules [Pearl, 1988]. The lower-level layer disambiguates individual letters while the higher-level layer disambiguates complete words (Figure 2). We assume that there are only two words that are expected to be seen, with equal probability: "THE" and "CAT".

There are three independent letter sensors with the middle sensor being unable to disambiguate the observed symbol /-\ represented by the conditional probabilities $p(H|/\!\!-\!\!\backslash) = 0.5$ and $p(A|/\!\!-\!\!\backslash) = 0.5$. These sensors feed into the lower-level nodes (or *processors* in Pearl's terminology), which we label as N_1, N_2, N_3. The results of the lower level nodes are combined at N_4 to disambiguate the observed word.

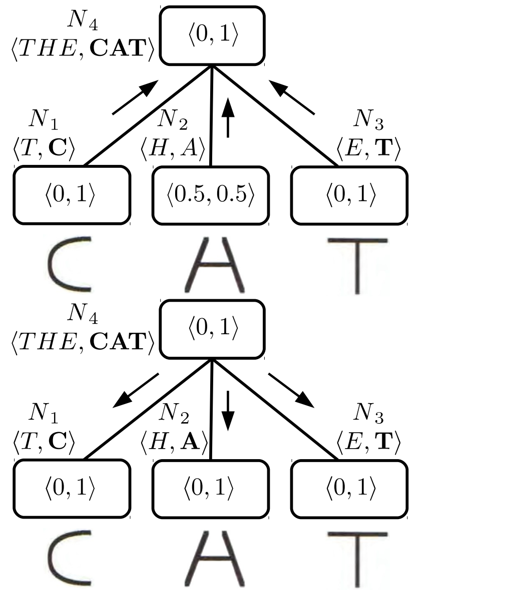

Figure 2: Disambiguating the symbol /-\ requires context from the word recognition layer.

Each node maintains two state variables; the *diagnostic* support and the *causal* support (displayed as the pairs of values in Figure 2). Intuitively, the diagnostic support represents the knowledge gathered through sensing while the causal support represents the contextual bias. A node's overall belief is calculated by the combination of these two state variables.

While sensing data propagates up the causal tree, the example highlights how node N_2 is only able to resolve the symbol /-\ in the presence of contextual feedback from node N_4.

2.2 Nodes

A cognitive hierarchy consists of a set of nodes. Nodes are tasked to achieve a goal or maximise future value. They have two primary functions: world-modelling and behaviour-generation. World-modelling involves maintaining a *belief state*, while behaviour-generation is achieved through *policies*, where a policy maps states to sets of actions. A node's belief state is modified either by sensing or by the combination of actions and higher-level context. We refer to this latter as *prediction update* to highlight how it sets an expectation about what the node is expecting to observe in the future.

Definition 1. *A cognitive language is a tuple* $\mathcal{L} = (\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{O}, \mathcal{C})$*, where* $\mathcal{S}$ *is a set of belief states,* $\mathcal{A}$ *is a set of actions,* $\mathcal{T}$ *is a set of task parameters,* $\mathcal{O}$ *is a set of observations, and* $\mathcal{C}$ *is a set of contextual elements. A cognitive node is a tuple* $N = (\mathcal{L}, \Pi, \lambda, \tau, \gamma, s^0, \pi^0)$ *s.t:*

- $\mathcal{L}$ *is the cognitive language for* N*, with initial belief state* $s^0 \in \mathcal{S}$.
- Π *a set of policies such that for all* $\pi \in \Pi$*,* $\pi : \mathcal{S} \to 2^{\mathcal{A}}$*, with initial policy* $\pi^0 \in \Pi$.
- *A policy selection function* $\lambda \colon 2^{\mathcal{T}} \to \Pi$*, s.t.* $\lambda(\{\}) = \pi^0$.
- *A observation update operator* $\tau : 2^{\mathcal{O}} \times \mathcal{S} \to \mathcal{S}$.

- *A prediction update operator* $\gamma : 2^{\mathcal{C}} \times 2^{\mathcal{A}} \times \mathcal{S} \to \mathcal{S}$.

Definition 1 differs from the original in two ways: the introduction of a set of context elements in the cognitive language, and the modification of the *prediction* update operator, previously called the *action* update operator, to include context elements when updating the belief state.

This definition can now be applied to the motivating example to instantiate the nodes in the Bayesian causal tree. We highlight only the salient features for this instantiation.

Example. *Let* $E = \{\langle x, y\rangle \mid 0 \leq x, y \leq 1.0\}$ *be the set of probability pairs, representing the recognition between two distinct features. For node* N_2*, say (cf. Figure 2), these features are the letters "H" and "A" and for* N_4 *these are the words "THE" and "CAT". The set of belief states for* N_2 *is* $\mathcal{S}_2 = \{\langle\langle d\rangle, c\rangle \mid d, c \in E\}$*, where* d *is the* diagnostic *support and* c *is the* causal *support. Note, the vector-in-vector format allows for structural uniformity across nodes. Assuming equal probability over letters, the initial belief state is* $\langle\langle\langle 0.5, 0.5\rangle\rangle, \langle 0.5, 0.5\rangle\rangle$*. For* N_4 *the set of belief states is* $\mathcal{S}_4 = \langle\langle d_1, d_2, d_3\rangle, c\rangle \mid d_1, d_2, d_3, c \in E\}$*, where* d_i *is the contribution of node* N_i *to the diagnostic support of* N_4.

For N_2 *the* context *is the causal supports from above,* $\mathcal{C}_2 = E$*, while the* observations *capture the influence of the "H"-"A" sensor,* $\mathcal{O}_2 = \{\langle d\rangle \mid d \in E\}$*. In contrast the observations for* N_4 *need to capture the influence of the different child diagnostic supports, so* $\mathcal{O}_4 = \{\langle d_1, d_2, d_3\rangle \mid d_1, d_2, d_3 \in E\}$.

The observation update operators need to replace the diagnostic supports of the current belief with the observation, which is more complicated for N_4 *due to its multiple children,* $\tau_2(\{\vec{d_1}, \vec{d_2}, \vec{d_3}\}, \langle\vec{d}, c\rangle) = \langle\Sigma_{i=1}^{3}\vec{d_i}, c\rangle$*. Ignoring the influence of actions, the prediction update operator simply replaces the causal support of the current belief with the context from above, so* $\gamma_2(\{c'\}, \emptyset, \langle\langle\vec{d}\rangle, c\rangle) = \langle\langle\vec{d}\rangle, c'\rangle$.

2.3 Cognitive Hierarchy

Nodes are interlinked in a hierarchy, where sensing data is passed up the *abstraction hierarchy*, while actions and context are sent down the hierarchy (Figure 3).

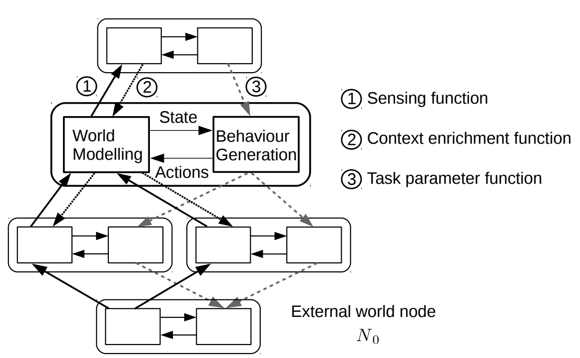

Figure 3: A cognitive hierarchy, highlighting internal interactions as well as the sensing, action, and context graphs.

Definition 2. *A cognitive hierarchy is a tuple* $H = (\mathcal{N}, N_0, F)$ *s.t:*

- $\mathcal{N}$ *is a set of cognitive nodes and* $N_0 \in \mathcal{N}$ *is a distinguished node corresponding to the external world.*
- F *is a set of function triples* $\langle\phi_{i,j}, \psi_{j,i}, \varrho_{j,i}\rangle \in F$ *that connect nodes* $N_i, N_j \in \mathcal{N}$ *where:*
  - $\phi_{i,j} : \mathcal{S}_i \to 2^{\mathcal{O}_j}$ *is a sensing function, and*
  - $\psi_{j,i} : 2^{\mathcal{A}_j} \to 2^{\mathcal{T}_i}$ *is a task parameter function.*
  - $\varrho_{j,i} : \mathcal{S}_j \to 2^{\mathcal{C}_i}$ *is a context enrichment function.*
- *Sensing graph: each* $\phi_{i,j}$ *represents an edge from node* N_i *to* N_j *and forms a directed acyclic graph (DAG) with* N_0 *as the unique source node of the graph.*
- *Prediction graph: the set of task parameter functions (equivalently, the context enrichment functions) forms a converse to the sensing graph such that* N_0 *is the unique sink node of the graph.*

Definition 2 differs from the original only in the introduction of the *context enrichment* functions and the naming of the *prediction graph* (originally the *action graph*). The connection between nodes consists of triples of sensing, task parameter and context functions. The sensing function extracts observations from a lower-level node in order to update a higher level node, while the context enrichment function performs the converse. The task parameter function translates a higher-level node's actions into a set of task parameters, which is then used to select the active policy for a node.

Finally, the external world is modelled as a distinguished node, N_0. Sensing functions allow other nodes to observe properties of the external world, and task parameter functions allow actuator values to be modified, but N_0 doesn't "sense" properties of other nodes, nor does it generate task parameters for those nodes. Similarly, context enrichment functions connected to N_0 would simply return the empty set, unless one wanted to model unusual properties akin to the quantum effects of observations on the external world. Beyond this, the internal behaviour of N_0 is considered to be opaque.

The running example can now be encoded formally as a cognitive hierarchy, again with the following showing only the salient features of the encoding.

Example. *We construct a hierarchy* $H = (\mathcal{N}, N_0, F)$*, with* $\mathcal{N} = \{N_0, N_1, \ldots, N_4\}$*. The function triples in* F *will include* $\phi_{0,2}$ *for the visual sensing of the middle letter, and* $\phi_{2,4}$ *and* $\varrho_{4,2}$ *for the sensing and context between* N_2 *and* N_4.

The function $\phi_{0,2}$ *returns the probability of the input being the characters "H" and "A". Here* $\phi_{0,2}(/\!\!-\!\!\backslash) = \{\langle 0.5, 0.5\rangle\}$.

Defining $\phi_{2,4}$ *and* $\varrho_{4,2}$ *requires a conditional probability matrix* $M = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ *to capture how the letters "H" and "A" contribute to the recognition of "THE" and "CAT".*

For sensing from N_2 *we use zeroed vectors to not influence the diagnostic support components from* N_1 *and* N_2*. Hence* $\phi_{2,4}(\langle\langle d\rangle, c\rangle) = \{\langle\langle 0, 0\rangle, \eta \cdot M \cdot d^T, \langle 0, 0\rangle\rangle\}$*, where* d^T *is the transpose of vector* d *and* η *is a normalisation constant.*

For context we capture how N_4*'s causal support and its diagnostic support components from* N_1 *and* N_2 *influences the causal support of* N_2*. Note, that this also prevents any feedback from* N_2*'s own diagnostic support to its causal support. So,* $\varrho_{4,2}(\langle\langle d_1, d_2, d_3\rangle, c\rangle) = \{\eta \cdot (d_1 \cdot d_3 \cdot c) \cdot M\}$.

2.4 Active Cognitive Hierarchy

The above definitions capture the static aspects of a system but require additional details to model its operational behaviour. Note, the following definitions are unmodified from the original formalism and are presented here because they are necessary to the developments of Section 2.5.

Definition 3. *An active cognitive node is a tuple $Q = (N, s, \pi, a)$ where: 1) N is a cognitive node with $\mathcal{S}$, Π, and $\mathcal{A}$ being its set of belief states, set of policies, and set of actions respectively, 2) $s \in \mathcal{S}$ is the current belief state, $\pi \in \Pi$ is the current policy, and $a \in 2^{\mathcal{A}}$ is the current set of actions.*

Essentially an active cognitive node couples a (static) cognitive node with some dynamic information; in particular the current belief state, policy and set of actions.

Definition 4. *An active cognitive hierarchy is a tuple $\mathcal{X} = (H, \mathcal{Q})$ where H is a cognitive hierarchy with set of cognitive nodes $\mathcal{N}$ such that for each $N \in \mathcal{N}$ there is a corresponding active cognitive node $Q = (N, s, \pi, a) \in \mathcal{Q}$ and vice-versa.*

The active cognitive hierarchy captures the dynamic state of the system at a time instance. Finally, an *initial active cognitive hierarchy* is an active hierarchy where each node is initialised with the initial belief state and policy of the corresponding cognitive node, as well as an empty set of actions.

2.5 Cognitive Process Model

The *process model* defines how an active cognitive hierarchy evolves over time and consists of two steps. Firstly, sensing observations are passed up the hierarchy, progressively updating the belief state of each node. Next, task parameters and context are passed down the hierarchy updating the active policy, the actions, and the belief state of the nodes.

We do not present all the definitions here, in particular we omit the definition of the *sensing update* operator, **SensingUpdate**, as this remains unchanged in our extension. Instead we define a *prediction update* operator, replacing the original *action update*, that incorporates both context and task parameters in its update. First, we characterise the updating of the beliefs and actions for a single active cognitive node.

Definition 5. *Let $\mathcal{X} = (H, \mathcal{Q})$ be an active cognitive hierarchy with $H = (\mathcal{N}, N_0, F)$. The prediction update of $\mathcal{X}$ with respect to an active cognitive node $Q_i = (N_i, s_i, \pi_i, a_i) \in \mathcal{Q}$, written as* $\textbf{PredUpdate}'(\mathcal{X}, Q_i)$ *is an active cognitive hierarchy $\mathcal{X}' = (H, \mathcal{Q}')$ where $\mathcal{Q}' = \mathcal{Q} \setminus \{Q_i\} \cup \{Q'_i\}$ and $Q'_i = (N_i, \gamma_i(C, a'_i, s_i), \pi'_i, a'_i)$ s.t:*

- *if there is no node N_x where $\langle \phi_{i,x}, \psi_{x,i}, \varrho_{x,i} \rangle \in F$ then: $\pi'_i = \pi_i$, $a'_i = \pi_i(s_i)$ and $C = \emptyset$,*
- *else:*

$$
\begin{aligned}
\pi'_i &= \lambda_i\,(T) \text{ and } a'_i = \pi'_i(s_i), \\
T &= \bigcup \{\psi_{x,i}(a_x) \mid \langle \phi_{i,x}, \psi_{x,i}, \varrho_{x,i} \rangle \in F \text{ where } \\
&\qquad Q_x = (N_x, s_x, \pi_x, a_x) \in \mathcal{Q}\} \\
C &= \bigcup \{\varrho_{x,i}(s_x) \mid \langle \phi_{i,x}, \psi_{x,i}, \varrho_{x,i} \rangle \in F \text{ where } \\
&\qquad Q_x = (N_x, s_x, \pi_x, a_x) \in \mathcal{Q}\}
\end{aligned}
$$

The intuition for Definition 5 is straightforward. Given a cognitive hierarchy and a node to be updated, the update process returns an identical hierarchy except for the update node. This node is updated by first selecting a new active policy based on the task parameters of all the connected higher-level nodes. The new active policy is applied to the existing belief state to generate a new set of actions. Both these actions and the context from the connected higher-level nodes are then used to update the node's belief state.

Using the single node update, updating the entire hierarchy simply involves successively updating all its nodes.

Definition 6. *Let $\mathcal{X} = (H, \mathcal{Q})$ be an active cognitive hierarchy with $H = (\mathcal{N}, N_0, F)$ and Ψ be the prediction graph induced by the task parameter functions in F. The action process update of $\mathcal{X}$, written* $\textbf{PredUpdate}(\mathcal{X})$*, is an active cognitive model:*

$$\mathcal{X}' = \textbf{PredUpdate}'(\ldots \textbf{PredUpdate}'(\mathcal{X}, Q_n), \ldots Q_0)$$

where the sequence $[Q_n, \ldots, Q_0]$ consists of all active cognitive nodes of the set $\mathcal{Q}$ such that the sequence satisfies the partial ordering induced by the prediction graph Ψ.

Importantly, the update ordering in Definition 6 satisfies the partial ordering induced by the prediction graph, thus guaranteeing that the prediction update is well-defined.

Lemma 1. *For any active cognitive hierarchy $\mathcal{X}$ the prediction process update of $\mathcal{X}$ is well-defined.*

Proof. Follows from the DAG. □

The final part of the process model, which we omit here, is the combined operator, **Update**, that first performs a sensing update followed by a prediction update. This operation follows exactly the original and similarly the theorem that the process model is well-defined also follows.

We can now apply the update process (sensing then prediction) to show how it operates on the running example.

Example. *When N_2 senses the symbol /-\, $\phi_{0,2}$ returns that "A" and "H" are equally likely, so τ_2 updates the diagnostic support of N_2 to $\langle\langle 0.5, 0.5 \rangle\rangle$. On the other hand N_1 and N_2 unambiguously sense "C" and "T" respectively, so N_4's observation update operator, τ_4, will update its diagnostic support components to $\langle \langle 0,1 \rangle, \langle 0.5, 0.5 \rangle, \langle 0, 1 \rangle \rangle$. The nodes overall belief, $\langle 0, 1 \rangle$, is the normalised product of the diagnostic support components and the causal support, indicating here the unambiguous recognition of "CAT".*

Next, during prediction update, context from N_4 is passed back down to N_2, through $\phi_{4,2}$ and γ_2, updating the causal support of N_2 to $\langle 0, 1 \rangle$. Hence, N_2 is left with the belief state $\langle \langle \langle 0.5, 0.5 \rangle \rangle, \langle 0, 1 \rangle \rangle$, which when combined, indicates that the symbol /-\ should be interpreted as an "A".

3 Causal Networks as Cognitive Hierarchies

The example in the previous section highlighted the use of context in a cognitive hierarchy inspired by belief propagation in causal trees. In this section we extend this example to the general result that any Bayesian causal tree can be encoded as a cognitive hierarchy. We do this by constructively showing how to encode a causal tree as a cognitive hierarchy and proving the correctness of this method with respect to propagating changes through the tree.

Pearl describes a causal tree as a set of *processors* where the connection between processors is explicitly represented within the processors themselves. Each processor maintains its diagnostic and causal support, as well as maintaining a conditional probability matrix for translating to the representation of higher-level processors. The description of the operational behaviour of causal trees is presented throughout Chapter 4 (and summarised in Figure 4.15) of [Pearl, 1988].

The cognitive hierarchies introduced here are concerned with robotic systems and consequently maintain an explicit notion of sensing over time. In contrast causal networks are less precise about external inputs and changes over time. As a bridge, we model that each processor has a diagnostic support component that can be set externally. Finally, note that we adopt the convenience notation $f_\emptyset$ to represent a function of arbitrary arity that always returns the empty set.

Definition 7. *Let $\{P_1, \dots P_n\}$ be a causal tree. We construct a corresponding cognitive hierarchy $H = (\{N_0, N_1, \dots, N_n\}, N_0, F)$ as follows:*

- *For processor P_i with m children, and diagnostic and causal supports $d, c \in \mathbb{R}^n$, define $\mathcal{S}_i = \{\langle\langle d_E, d_1, \dots, d_m\rangle, c'\rangle | d_E, d_1, \dots, d_m, c' \in \mathbb{R}^n\}$, with initial belief state $s_i = \langle\langle d, \dots, d\rangle, c\rangle$. Define $\mathcal{O}_i = \{\langle d_E, d_1, \dots, d_m\rangle | d_E, d_1, \dots, d_m \in \mathbb{R}^n\}$ and $\mathcal{C}_i = \mathbb{R}^n$.*
- *For processor P_i with corresponding cognitive node N_i, define $\tau_i(o, \langle\vec{d}, c\rangle) = \langle\Sigma_{\vec{d'} \in o}\vec{d'}, c\rangle$, and $\gamma_i(\{c'\}, \emptyset, \langle\vec{d}, c\rangle) = \langle\vec{d}, c'\rangle$.*
- *For each pair of processors P_i and P_j, where P_j is the k-th child of P_i's m children (from processor subscript numbering), and M_j is the conditional probability matrix of P_j, then define a triple $\langle\phi_{j,i}, f_\emptyset, \varrho_{i,j}\rangle \in F$ s.t:*
  - *$\phi_{j,i}(\langle\vec{d}, c\rangle) = \{\langle d_E, d_1, \dots, d_m\rangle\}$, where $d_{h \neq k}$ are zeroed vectors and $d_k = \eta \cdot M_j \cdot (\prod_{d' \in \vec{d}} d')^T$.*
  - *$\varrho_{i,j}(\langle\langle d_E, d_1, \dots, d_m\rangle, c\rangle) = \{c'\}$, such that $c' = \eta \cdot (\prod_{h \neq k} d_h \cdot c) \cdot M_j$.*
  - *where η is a normalisation constant for the respective vectors, and x^T is the transpose of vector x.*
- *For processors P_i, with diagnostic support $d \in \mathbb{R}^n$, define a triple $\langle\phi_{0,i}, f_\emptyset, f_\emptyset\rangle \in F$ where $\phi_{0,i}(s_0 \in \mathcal{S}_0) = \{\langle d_E, d_Z, \dots, d_Z\rangle\}$, where d_Z is a zeroed vector and $d_E \in \mathbb{R}^n$ is the external input of P_i.*

While notationally dense, Definition 7 is a generalisation of the construction used in the running example and is a direct encoding of Pearl's causal trees. This construction could be further extended to poly-trees, which Pearl also considers, but would require a slightly more complex encoding.

To establish the correctness of this transformation we can compare how the structures evolve with sensing. The belief measure of a processor is captured as the normalised product of the diagnostic and causal supports, $BEL(P_i) = \eta \cdot d_i \cdot c_i$. However, for a cognitive node the diagnostic support needs to be computed from its components. Hence, given the belief state $\langle\langle d_E, d1, \dots, d_m\rangle, c\rangle$ of an active node Q_i with m children, we can compute the belief as $BEL(Q_i) = \eta \cdot \prod_{j=1}^{m} d_j \cdot c$.

Lemma 2. *Given a causal tree $\{P_1, \dots P_n\}$ and a corresponding cognitive hierarchy H constructed via Definition 7, then the causal tree and the initial active cognitive hierarchy corresponding to H share the same belief.*

Proof. By inspection, $BEL(P_i) = BEL(Q_i)$ for each i. □

Now, we establish that propagating changes through an active cognitive hierarchy is consistent with propagating beliefs through a causal tree. We abuse notation here to express the overall belief of a casual tree (resp. active cognitive hierarchy) as simply the beliefs of its processors (resp. nodes).

Theorem 3. *Let $\mathcal{T}$ be a causal tree and $\mathcal{X}$ be the corresponding active cognitive hierarchy constructed via Definition 7, such that $BEL(\mathcal{T}) = BEL(\mathcal{X})$. Then for any changes to the external diagnostic supports of the processors and corresponding changes to the sensing inputs for the active cognitive hierarchy, $BEL(Prop(\mathcal{T})) = BEL(\textbf{Update}(\mathcal{X}))$.*

Proof. Pearl establishes that changes propagated through a causal tree converge with a single pass up and down the tree. Any such pass satisfies the partial ordering for the cognitive hierarchy process model. Hence the proof involves the iterative application of the process model, confirming at each step that the beliefs of the processors and nodes align. □

Theorem 3 establishes that Bayesian causal trees can be captured as cognitive hierarchies. This highlights the significance of extending cognitive hierarchies to include context, allowing for a richer set of potential applications.

4 Using Context to Track Objects Visually

Object tracking has application in augmented reality, visual servoing, and man-machine interfaces. We consider the problem of on-line monocular model-based tracking of multiple objects without markers or texture, using the monocular camera built into the hand of a Baxter robot. The use of natural object features makes this a challenging problem.

A basic approach to tackling this problem is to use 3D contextual knowledge in the form of a CAD model, from which to generate a set of edge points (control points) for the object [Lepetit and Fua, 2005] . The idea is to track the corresponding 2D camera image points of the visible 3D control points as the object moves relatively to the camera. The new pose of the object relative to the camera is found by minimising the perspective re-projection error between the control points and their corresponding 2D image

However, when multiple objects are tracked, independent CAD models fail to handle object occlusion. We replace the CAD models by the machinery provided by a 3D physics simulator. The object-scene and virtual cameras from a simulator are ideal to model the higher level context for vision. We now describe how this approach is instantiated as a cognitive hierarchy with contextual feedback. It is important to note that the use of the physics simulator is not to replace the real-world, but is used as mental imagery efficiently representing the spatial belief state of the robot.

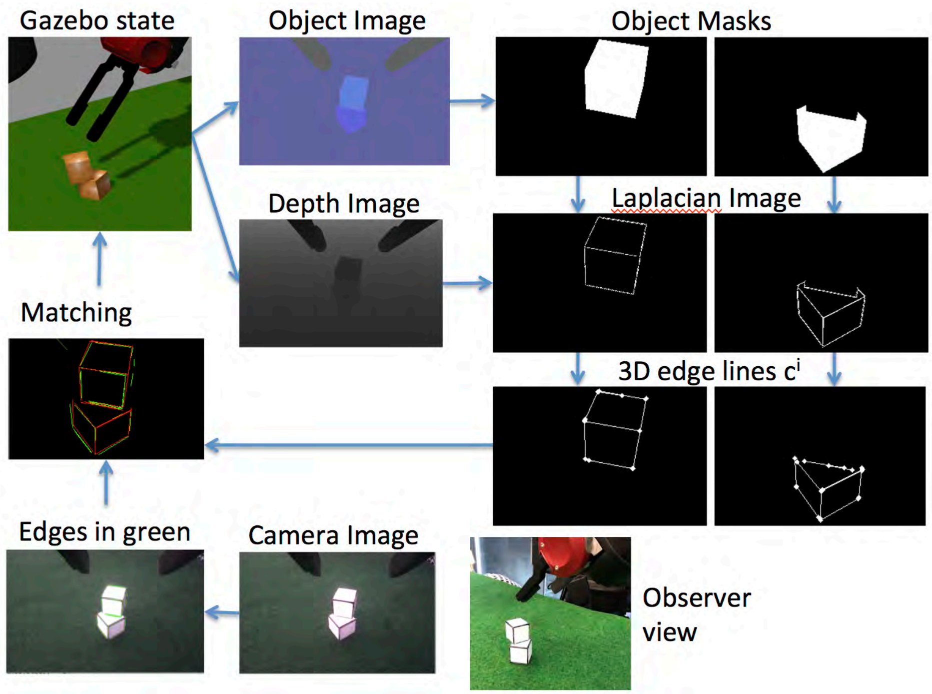

Figure 4: The Process update showing stages of the context enrichment function and the matching of contextual information to the real camera to correct the arm and spatial node belief state.

4.1 Cognitive Hierarchy for Visual Tracking

We focus on world-modelling in a two-node cognitive hierarchy (Figure 5). The external world node that includes the Baxter robot, streams the camera pose and RGB images as sensory input to the arm node. The arm node belief state $s = \{p^a\} \cup \{\langle p_a^i, c^i \rangle | \text{object } i\}$, where p^a is the arm pose, and for all recognised objects i in the field of view of the arm camera, p_a^i is the object pose relative to the arm camera, and c^i is the set of object edge lines and their depth. The objects in this case include small cubes on a table. Information from the arm node is sent to the spatial node that employs a Gazebo physics simulator as mental imagery to model the objects.

A novel feature of the spatial node is that it simulates the robot's arm camera as a depth camera, underlining its spatial understanding of the scene. The expected object surfaces visible to the real camera, segmented into depth clouds by object, are passed to the arm node. In turn it uses this contextual data to adjust poses, and thus track the objects in view.

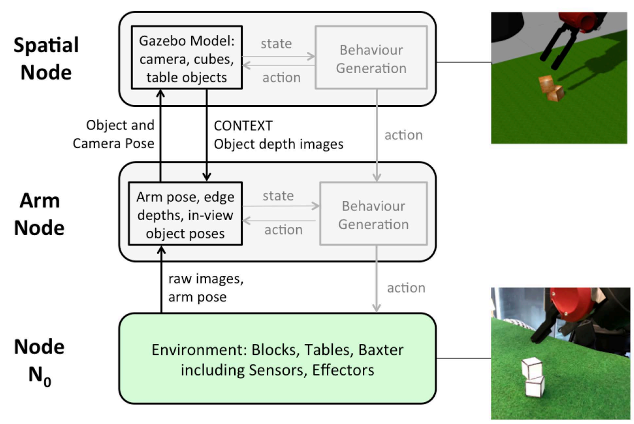

Figure 5: Cognitive hierarchy comprising an arm node and a spatial node. Context from the spatial node is an object segmented depth image from a simulation of the real camera.

4.2 Update Functions and Process Update

We now describe the update functions and a single cycle of the process update for this cognitive hierarchy.

The real monocular RGB arm camera is simulated in Gazebo with an object aware depth camera with identical characteristics (i.e. the same intrinsic camera matrix). The simulated camera then produces depth and an object segmentation images from the simulated objects that corresponds to the actual camera image. This vital contextual information is then used for correcting the pose of the visible objects.

The process update starts with the sensing function $\phi_{N_0,Arm}$ taking the raw camera image and observing all edges in the image represented as a set of line segments, l.

$$\phi_{N_0,Arm}(\{rawImage\}) = \{l\}$$

The observation update operator τ_{Arm} takes the expected edge lines c^i for each object i and transforms the lines to best match the image edge lines l. The update uses an ICP-like algorithm to find a corrected pose p_a^i for each object i relative to the arm-camera a [2].

$$\tau_{Arm}(\{l, c^i|\text{object } i\}) = \{p_a^i|\text{object } i\}$$

The sensing function from the arm to spatial node takes the corrected pose p_a^i for each object i, relative to the camera frame a, and transforms it into the Gazebo reference frame via the Baxter's reference frame given the camera pose p^a.

$$\phi_{Arm,Spatial}(\{p^a, \langle p_a^i, c^i\rangle|\text{object } i\}) = \{g_a^i|\text{object } i\}$$

The spatial node observation update $\tau_{Spatial}$, updates the pose of all viewed objects g_a^i in the Gazebo physics simulator. Note $\{g_a^i|\text{object } i\} \subset$ gazebo state.

$$\tau_{Spatial}(\{g_a^i|\text{object } i\}) = \text{gazebo.move}(i, g_a^i) \ \ \forall i$$

The update cycle now proceeds down the hierarchy with prediction updates. The prediction update for the spatial node $\gamma_{Spatial}$ consists of predicting the interaction of objects in the simulator under gravity. Noise introduced during the observation update may result in objects separating due to detected collisions or settling under gravity.

$$\gamma_{Spatial}(\text{gazebo state}) = \text{gazebo.simulate}(\text{gazebo state}))$$

We now turn to the context enrichment function $\varrho_{Spatial,Arm}$ that extracts predicted camera image edge lines and depth data for each object in view of the simulator.

$$\varrho_{Spatial,Arm}(\text{gazebo state}) = \{c^i|\text{object } i\}$$

The stages of the context enrichment function $\varrho_{Spatial,Arm}$ are shown in Figure 4. The simulated depth camera extracts an object image that identifies the object seen at every pixel location. It also extracts a depth image that gives the depth from the camera of every pixel. The object image is used to mask out each object in turn. Applying a Laplacian function to the part of the depth image masked out by the object yields all visible edges of the object. A Hough line transform identifies line end points in the Laplacian image and finds the depth of their endpoints from the depth image, producing c^i.

Figure 6 highlights how the cognitive hierarchy tracks cubes in the face of object and arm camera movements.

5 Discussion

There is considerable evidence supporting the existence and usefulness of top-down contextual information. For example, incongruent elements in an image are recognised less reliably, demonstrating top-down analyses from the content of a scene [Biederman *et al.*, 1981]. Cavanagh [1991] showed that top-down processing speeds the analysis of the retinal image when familiar scenes and objects are encountered. In cognitive psychology this is related to the *context effect*, where environmental factors influence perception, and *constructive perception* where other top-down sources of information construct a cognitive understanding of the sensory stimulation.

These observations are further supported by neuroscience, suggesting that feedback pathways from higher more abstract processing areas of the brain down to areas closer to the sensors are greater than those transmitting information upwards [Hawkins and Blakeslee, 2004]. The authors summarise the process - "what is actually happening flows up, and what you expect to happen flows down". Gilbert and Li [2013] argue that the traditional idea that the processing of visual information consists of a sequence of feedforward operations needs to be supplemented by top-down contextual influences.

Context is consistent with the Gestalt theory of perception that posits that we understand phenomena by viewing them as organised and structured wholes rather than the sum of their constituent parts. Earlier preliminary experiments showed how a visual perceptual hierarchy built up from edgelets, and including contextual exceptions of seeing either squares, triangles or circles, could see for example a Kanizsa triangle 7.

In the field of robotics, recent work in online interactive perception also shows the benefit of predicted measurements from one level being passed to the next-lower level as state predictions [Martin and Brock, 2014].

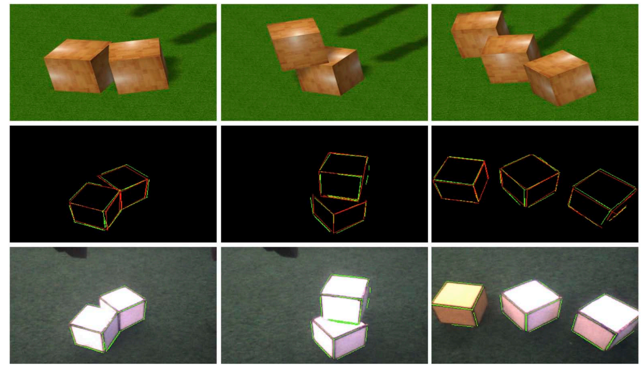

Figure 6: Tracking several cube configurations. Top row: Gazebo GUI showing spatial node state. 2nd row: matching real image edges in green to simulated image edges in red. Bottom row: camera image overlaid with edges in green.

6 Future Work

The cognitive hierarchy [Clark *et al.*, 2016], now with the addition of context, is being further developed in two ways:

Behaviour Utility Behaviour generation is currently formalised in the cognitive hierarchy as a top-down process, where more abstract nodes select the action policy of less abstract nodes. To guide this selection process, the more abstract nodes need access to the cost or utility of the options available for selection if they are to choose better behaviours. It is in the agents interest to not just find a satisficing solution, but a cost-effect, if not optimal, solution.

To achieve this functionality, we require utility information to be passed up the behaviour generation hierarchy so that a value function can be composed to allow for more rational choice of actions. The idea is

[2] The pose of a rigid object in 3D space has 6 degrees of freedom, three describing its translated position, and three the rotation or orientation, relative to a reference pose.

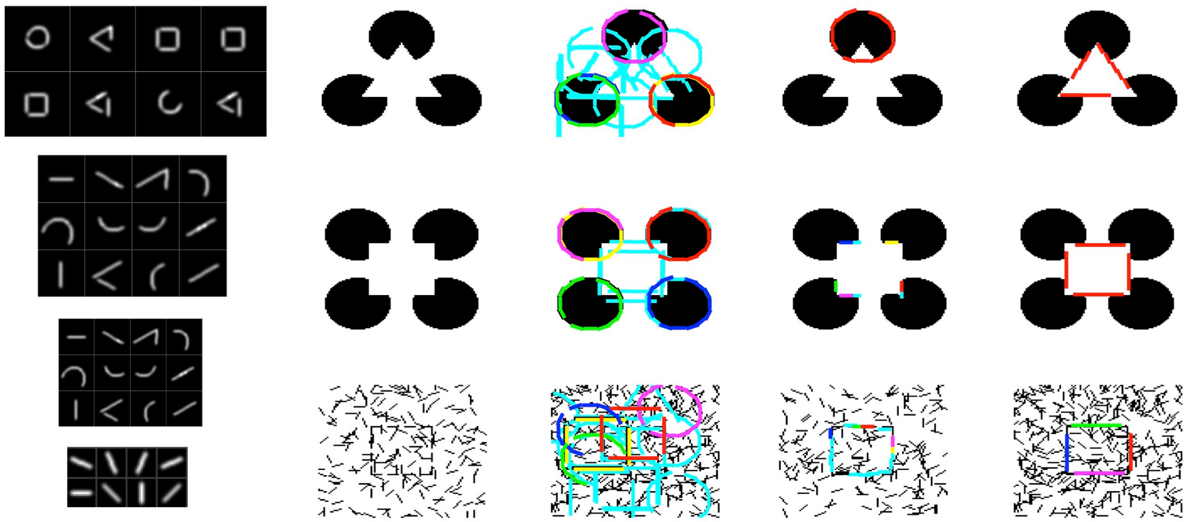

Figure 7: A perceptual hierarchy showing that contextual triangles, circles and squares can be perceived as illusionary contours in images such as those created by Gaetano Kanizsa (1955), or in clutter. The perceptual hierarchy progressively composes edgelet features into more complex shapes culminating in the triangles, circles, and squares. The column of images on the left side of the figure shows progressively more complex shapes learned from primitive edgelet features. In the rest of the figure the top row shows both a circle shape and a triangle shape recognised in the three-“Pac-Man” like image. The middle row shows a similar phenomenon for a square. The bottom set of images show a square recognised in extreme clutter of random edgelets. Source - unpublished experiments from Nobuyuki Morioka and Bernhard Hengst 2009.

to extend the value function recomposition from hierarchical reinforcement learning [Dietterich, 2000; Hengst, 2002] to our framework that integrates symbolic and sub-symbolic representations.

Learning The description of the cognitive hierarchy has been silent on learning the various world model maintenance and behaviour generation functions. It is our intention to include learning as a capability. As an example, reinforcement learning suggests how the prediction update operator, i.e. the state transition function, and the policy function can be learned, and for the system to improve its performance over time. Several anytime schemes can be used to choose good action policies given limited resource constraints such as time.

The challenge is to instantiate cognitive hierarchies capable of developmental behaviour generation to thrive in a particular environment over the life-time of the agent.

7 Conclusion

This paper formalises the notion contextual feedback in a cognitive hierarchy, interprets Pearl's belief updating in causal trees as such a hierarchy and demonstrates the importance of context in a challenging vision task. We believe the notion of context and its influence will play a larger role in robotics and artificial intelligence research.

Acknowledgments

This material is based upon work supported by the Asian Office of Aerospace Research and Development (AOARD) under Award No: FA2386-15-1-0005. This research was also supported under Australian Research Council's (ARC) *Discovery Projects* funding scheme (project number DP 150103035). Michael Thielscher is also affiliated with the University of Western Sydney.

We also thank our anonymous IJCAI 2017 and AGA 2017 reviewers for their insightful and helpful comments on earlier versions of this paper.

Disclaimer

Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the AOARD.

References

[Albus and Meystel, 2001] James S. Albus and Alexander M. Meystel. *Engineering of Mind: An Introduction to the Science of Intelligent Systems*. Wiley-Interscience, 2001.

[Ashby, 1952] W. Ross Ashby. *Design for a Brain*. Chapman and Hall, 1952.

[Bakker and Schmidhuber, 2004] Bram Bakker and J urgen Schmidhuber. Hierarchical reinforcement learning based on subgoal discovery and subpolicy specialization. In *Proceedings of the 8-th Conference on Intelligent Autonomous Systems, IAS-8*, pages 438–445, 2004.

[Beer, 1966] Stafford Beer. *Decision and Control*. John Wiley and Sons, London, 1966.

[Biederman *et al.*, 1981] I Biederman, M Kubovy, and JR Pomerantz. On the semantics of a glance at a scene. In *Perceptual Organization*, pages 213–263. Lawrence Erlbaum, New Jersey, 1981.

[Brooks, 1986] Rodney A. Brooks. A robust layered control system for a mobile robot. *Robotics and Automation, IEEE Journal of*, 2(1):14–23, Mar 1986.

[Cavanagh, 1991] P. Cavanagh. What's up in top-down processing? In A. Gorea, editor, *Representations of Vision: Trends and tacit assumptions in vision research*, number 295-304, 1991.

[Clark *et al.*, 2016] Keith Clark, Bernhard Hengst, Maurice Pagnucco, David Rajaratnam, Peter Robinson, Claude Sammut, and Michael Thielscher. A framework for integrating symbolic and sub-symbolic representations. In *25th Joint Conference on Artificial Intelligence (IJCAI - 16)*, 2016.

[Dayan and Hinton, 1992] Peter Dayan and Geoffrey E. Hinton. Feudal reinforcement learning. *Advances in Neural Information Processing Systems 5 (NIPS)*, 1992.

[Dietterich, 2000] Thomas G. Dietterich. Hierarchical reinforcement learning with the MAXQ value function decomposition. *Journal of Artificial Intelligence Research (JAIR)*, 13:227–303, 2000.

[Drescher, 1991] Gary L. Drescher. *Made-up Minds: A constructionist Approach to Artificial Intelligence*. MIT Press, Cambridge, Massachusetts, 1991.

[Gilbert and Li, 2013] Charles D Gilbert and Wu Li. Top-down influences on visual processing. *Nature reviews. Neuroscience*, 14(5):10.1038/nrn3476, 05 2013.

[Hawkins and Blakeslee, 2004] Jeff Hawkins and Sandra Blakeslee. *On Intelligence*. Times Books, Henry Holt and Company, 2004.

[Hengst *et al.*, 2016] Bernhard Hengst, Nadine Marcus, Maurice Pagnucco, David Rajaratnam, Claude Sammut, and Michael Thielscher. Towards autonomous adaptation and trust. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2016) - Tenth International Cognitive Robotics Workshop*, 2016.

[Hengst, 2002] Bernhard Hengst. Discovering hierarchy in reinforcement learning with HEXQ. In Claude Sammut and Achim Hoffmann, editors, *Proceedings of the Nineteenth International Conference on Machine Learning*, pages 243–250. Morgan-Kaufman, 2002.

[Hubel and Wiesel, 1979] David H. Hubel and Torsten N. Wiesel. Brain mechanisms of vision. *A Scientific American Book: the Brain*, pages 84–96, 1979.

[Johnson, 2010] Jeff Johnson. *Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Rules*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2010.

[Jong, 2010] Nicholas K. Jong. *Structured Exploration for Reinforcement Learning*. PhD thesis, University of Texas at Austin, 2010.

[Kaelbling, 1993] Leslie Pack Kaelbling. Hierarchical learning in stochastic domains: Preliminary results. In *Machine Learning Proceedings of the Tenth International Conference*, pages 167–173, San Mateo, CA, 1993. Morgan Kaufmann.

[Konidaris *et al.*, 2011] George Konidaris, Scott Kuindersma, Roderic Grupen, and Andrew Barto. Robot learning from demonstration by constructing skill trees. *The International Journal of Robotics Research*, 2011.

[Lepetit and Fua, 2005] Vincent Lepetit and Pascal Fua. Monocular model-based 3d tracking of rigid objects. *Found. Trends. Comput. Graph. Vis.*, 1(1):1–89, January 2005.

[Marthi *et al.*, 2006] Bhaskara Marthi, Stuart Russell, and David Andre. A compact, hierarchical q-function decomposition. In *Proceedings of the Proceedings of the Twenty-Second Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-06)*, pages 332–340, Arlington, Virginia, 2006. AUAI Press.

[Martin and Brock, 2014] Roberto Martin Martin and Oliver Brock. Online interactive perception of articulated objects with multi-level recursive estimation based on task-specific priors. In *IROS*, pages 2494–2501. IEEE, 2014.

[Minsky, 1986] Marvin Minsky. *The Society of Mind*. Simon & Schuster, Inc., New York, NY, USA, 1986.

[Nilsson, 2001] Nils J. Nilsson. Teleo-Reactive programs and the triple-tower architecture. *Electronic Transactions on Artificial Intelligence*, 5:99–110, 2001.

[Pearl, 1988] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Francesco, revised second printing edition, 1988.

[Rajaratnam *et al.*, 2016] David Rajaratnam, Bernhard Hengst, Maurice Pagnucco, Claude Sammut, and Michael Thielscher. Composability in cognitive hierarchies. In *AI 2016: Advances in Artificial Intelligence 29th Australasian Joint Conference*, volume 9992, pages 42–55. Springer, 2016.

[Turchin, 1977] Valentin Fedorovich Turchin. *The Phenomenon of Science*. Columbia University Press, 1977.