

Comparative Reasoning for Intelligent Agents

Patrick Hammer[1(✉)], Peter Isaev[2], Hugo Latapie[3], Francesco Lanza[4], Antonio Chella[4], and Pei Wang[2]

[1] Department of Psychology, Stockholm University, Stockholm, Sweden
patrick.hammer@psychology.su.se
[2] Department of Computer Science, Temple University, Philadelphia, USA
{peter.isaev,pei.wang}@temple.edu
[3] Cisco Inc., San Jose, USA
hlatapie@cisco.com
[4] Robotics Lab, Università degli Studi di Palermo, Palermo, Italy
{francesco.lanza,antonio.chella}@unipa.it

Abstract. We demonstrate new comparative reasoning abilities of NARS, a formal model of intelligence, which enable the asymmetric comparison of perceivable quantifiable attributes of objects using relations. These new abilities are implemented by extending NAL with additional inference rules. We demonstrate the new capabilities in a bottle-picking experiment on a mobile robot running ONA, an implementation of NARS.

Keywords: Non-Axiomatic Logic · Comparative Relation · Comparative Reasoning · Inference Rules · Visual object comparison · NARS

1 Introduction

Comparative reasoning has been extensively studied with animals. It has been demonstrated already decades ago, that various animal species can not only condition on concrete stimuli, but also on comparative relations between them, a capability referred to as Transposition [10]. These comparative relations can be about any perceivable properties objects possess, such as their color, size, or shape, or the loudness or pitch of the sounds they make. For most animals, as it turned out, such comparisons are trivial to make: they can condition on such relations almost instantly from just a few examples, and can apply them to novel objects they have never seen before. In computer systems only some aspects of these capabilities have been replicated, such as by deciding which code path to take in a program, based on the outcome of comparisons between numbers with mathematical operators in various programming languages. However, the difficulty lies not in number comparison itself, but in deciding what to compare, and when to compare, and also that most comparisons to be made are relative to another object or category and are hence context-dependent. Furthermore, to

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
P. Hammer et al. (Eds.): AGI 2023, LNAI 13921, pp. 126–135, 2023.
https://doi.org/10.1007/978-3-031-33469-6_13

exploit the properties of comparative relations, such as to derive new comparative relations from previous ones, reasoning is required. This is especially the case for asymmetric relations, which cannot be handled in a purely associative manner.

Based on our previous results related to fuzzy quantities and preferences [13,14], this paper extends Non-Axiomatic Logic (henceforth NAL, [15,16]) to support the formation and usage of comparative relations.

This capability has been added in ONA [7], an implementation of NARS (Non-Axiomatic Reasoning System), which will be demonstrated in an experiment with a mobile robot with manipulator arms.

2 Related Work

A representative attempt to support a cognitively plausible treatment of quantities attached to concepts is Conceptual Spaces [5]. Conceptual Spaces are vector spaces of semantically meaningful measures, with a pre-defined similarity metric, such as the color space represented as a three-dimensional space of red, green, and blue. The following represents a series of works in which conceptual spaces were used:

In [9], authors claim that the Conceptual Space can be used as a *lingua franca* for the different levels of representation. With Conceptual Space it becomes easy to unify and generalize many aspects of symbolic, diagrammatic and subsymbolic approaches and integrate them on a common ground. Various applications were realized using conceptual spaces.

In [4], Conceptual Spaces were used to define a framework for endowing a computer vision architecture with a high-level representation capability through the definition of conceptual semantics for the symbolic representations of the vision system. Another usage [3], for example, involves Conceptual Spaces to allow robotic systems to learn effectively by imitation, and not simply reproducing the movements of a human teacher. In that way, the system should be able to deeply understand the perceived actions to be imitated. In [2], the author presented a cognitive architecture for a musical agent. In this architecture, conceptual spaces were used to represent the perception of tones and intervals.

In contrast to these approaches, as we will see, NARS does not utilize predefined attribute vectors, and the similarity evaluation in our approach is dependent on previously experienced values of the compared attribute in a reference class, which supports to handle context-dependence in comparative reasoning, and more significantly, to find useful subsets of properties to compare with beyond what a human designer pre-specified.

Another less similar but still related approach is to make use of Deep Neural Network (DNN) or simpler function approximators to learn task-relevant relational comparison functions (as in [11]). This can also be combined with Reinforcement Learning (by learning a suitable DNN policy [17]).

Hereby, sample efficiency decreases drastically when task-relevant functions able to carry out value comparisons have to be learned from a blank slate while the agent is operating, which is avoided when having domain-independent comparative reasoning abilities inbuilt in the agent. Related evolved general-purpose

capabilities which establish data-efficient learning in survival-critical settings are long being studied [10].

3 Formalization of Comparative Relation

NARS is a model of intelligence that has been partially formalized and implemented [15,16]. It uses a concept-centered knowledge representation, in which a *concept* is a data structure summarizing a fragment or pattern of the system's experience, identified by a *term* in a formal language Narsese. A term is an internal name that can be atomic (i.e., an ID or key) or compound (i.e., a structure of other terms).

Some terms correspond to conceptual relations. A *Comparative Relation* is a special type of relation with two components for which a measurement or total order among a class of terms can be obtained. This is formalized as follows:

- Given a measurement M defined on a class C, each instance c of the class has a value v. This measurement can be written as $v = M(c)$, or in Narsese as $(\{c\} \times \{v\}) \rightarrow \uparrow M$, which intuitively means that M is an executable operation of the system that, when c is given as input, returns v as output.
- For any pair of instances of the class c_1 and c_2, their values can be compared to get one of the three possible results: $M(c_1) > M(c_2)$, $M(c_1) < M(c_2)$, or $M(c_1) = M(c_2)$. If M is not a measurement but a total order, the result is directly obtained.
- Using Boolean operators, disjunction and negation, $\geq$, $\leq$, and $\neq$ can also be obtained from the comparison in the usual way.
- Each of the comparisons of measurement results corresponds to a *comparative relation* between c_1 and c_2, such as $M(c_1) > M(c_2)$ can be written as $(\{c_1\} \times \{c_2\}) \rightarrow >M$, where the relation is identified by the measurement M with the comparative relation $>$ as a prefix.
- For each measurement, six such comparative relations can be defined. Here, the arithmetic comparison is interpreted or instantiated into the concrete relationship, that is, while $>$ compares two numbers, >LENGTH compares the lengths of two objects.

Relative Ranking

The previous works [13,14] have extended the comparisons from between two instances to between an instance and a reference class. Formally, $(\{c_1\} \times C) \rightarrow R$ is taken as a summary of $(\{c_1\} \times \{x\}) \rightarrow R$ for every x in C that has been compared to c_1, directly or indirectly. Therefore, in ideal situations where there is no uncertainty in the instance comparisons, the truth-value of $(\{c_1\} \times C) \rightarrow$ >LENGTH indicates its relative ranking in C on length, that is, the *frequency* of the statement indicates the percent of instances of C that c is longer than, while the *confidence* of the statement indicates the number of instances compared so far, relative to a constant.

In the simplest situations, such a conclusion is derived by the *induction* rule from $(\{c_1\} \times \{c_2\}) \rightarrow R$ and $\{c_2\} \rightarrow C$, with conversions between *product* and *image*. Such judgments are merged by the *revision* rule to get the statistical result incrementally. The final conclusion is however not necessarily purely statistical or inductive, as matching conclusions can be obtained via other inference paths [16]. Using the same approach, it is also possible to draw comparative conclusions between two reference classes (or generic terms) like $(C_1 \times C_2) \rightarrow R$.

When the measurement values are directly available (such as in a sensation), it is possible to take the difference of the measurement values $|v_1 - v_2|$ as the weight of evidence for each comparison, so the truth-value is not merely a relative measurement [14]. For example, if the available values are 1, 2, and 9, the *frequency* for 2 to be considered as "larger than the others" is not $1/(1 + 1)$, but $1/(1 + 7)$, as the negative evidence is much "heavier" than the positive evidence.

Comparative Property

As analyzed in [13], many "fuzzy" property or membership is caused by comparisons with instances in an implicit reference class.

For example, "The Amazon River is long" actually means "The Amazon River is longer than the other rivers", so in Narsese it should be $(\{Amazon\} \times river) \rightarrow$ >LENGTH or $\{Amazon\} \rightarrow$ (>LENGTH / _, $river$), rather than as $\{Amazon\} \rightarrow [long]$. This is the case because the "comparative property" $[long]$ is highly context-dependent, and largely determined by the reference class.

Another special feature of comparative relation appears in a statement like $\{Amazon\} \rightarrow$ (>LENGTH / _, $river$) where its *frequency* is low (near 0) or inconclusive (near 0.5). For ordinary *inheritance* judgments, low frequency usually means "no inheritance". In particular, if two objects both lack a property, such as "not red", they should not be considered similar to each other for this reason, however, for continuous measurements, close values provide evidence for their similarity, and the closer, the more evidence there is, no matter where they are in the range of the values. For example, if two rivers are "not long", then they are both "short", so are similar for that reason.

Also, in sensory channels the reference class is often implied. Instead of $(\{c_1\} \times Brightness) \rightarrow >bright$, one can write $\{c_1\} \mapsto [bright]$ as a shortcut.

Inference Rules

To provide inference on comparative relations, the following inference rules are added as an extension of NAL:

– Comparing two measurements:

$$\{(\{c_1\} \times \{v_1\}) \rightarrow \uparrow M,\ (\{c_2\} \times \{v_2\}) \rightarrow \uparrow M\} \vdash (\{c_1\} \times \{c_2\}) \rightarrow R$$

whereby R is either $<M$, $>M$, or $=M$ dependent on whether $v_1 < v_2$, $v_1 > v_2$, or $v_1 = v_2$ holds.

– Transitivity of comparative relation, using f_{ded} truth function [16]:

$$\{(A \times B) \rightarrow R,\ (B \times C) \rightarrow R\} \vdash (A \times C) \rightarrow R,\ \ f_{ded}$$

– Symmetry of comparative relations of type =:

$$\{(A \times B) \rightarrow =M\} \vdash (B \times A) \rightarrow =M$$

– Inversion of a comparative relation of type <and>:

$$\{(A \times B) \rightarrow >M\ \} \vdash (B \times A) \rightarrow <M$$

$$\{(A \times B) \rightarrow <M\ \} \vdash (B \times A) \rightarrow >M$$

– Exclusiveness of comparative relations:

$$\forall R, S \in \{<M, >M, =M\} : (R \neq S) \implies \{(A \times B) \rightarrow R\} \vdash \neg((A \times B) \rightarrow S)$$

– Negation of a comparative relation/property:

$$\forall R, S, T \in \{<M, >M, =M\} : ((R \neq S) \wedge (R \neq T)) \implies$$
$$\neg(A \times B) \rightarrow R\} \vdash (A \times B) \rightarrow S \vee (A \times B) \rightarrow T$$

– Inference rule and truth function for class-based comparison:

$$\{(\{c_1\} \times C) \rightarrow R,\ (\{c_2\} \times C) \rightarrow R\} \vdash (\{c_1\} \times \{c_2\}) \rightarrow R$$

whereby R is either $<M$, $>M$, dependent on whether $f_1 < f_2$, or $f_1 > f_2$ holds. This is similar as in value comparison, but please note that here we compare the revised frequency values relative to the class, which indicates what proportion of instances in that class the instances' M property is smaller or greater than. The conclusion confidence hereby is dependent on the premise confidences via $c = c_1 * c_2$, and the conclusion frequency is $f = 1$.

Comparisons via Commonalities and Differences

In perception, it is not feasible to exhaustively compare all encountered objects with all currently perceived and remembered ones, especially if each comparison needs to select an informative subset of properties as the basis for the comparison, which will hopefully be useful to condition on to achieve goals. Which brings us back to the question: what to compare with, and when? According to selective perception, active relational goals should request/trigger the task-relevant comparisons of interest, a form of top-down attention as discussed in [8]. However, how did the goal structure which brings them into existence emerge in the first place? One answer could be (and is the one we find most promising and choose to pursue) that attending to commonalities and differences between observed and remembered instances significantly reduces the number of possible groups of properties to consider for comparison. When an instance is recalled

from memory that is closest to the newly perceived instance, it is not uncommon that there will be dozens of matching properties (it was the closest instance after all). Now, any differences to the remembered instance are highly informative and might indicate a change, which allows categorizing the newly observed instance not only based on the closest memory item, but also according to the most significant difference to it, which is a much more concise description (as it references the closest, yet keeps the significant differences) than when having to include all the properties of the new instance in its encoding. We will later see the relevance of this in the experiment, but for this to work the following additional inference rules are required:

1. to obtain instance similarity from comparative properties with value-wise close measurements (e.g. to find the best-matching instance in memory):

$$\{\{c_1\} \rightarrow (>\text{M} \;/\; _, C),\ \{c_2\} \rightarrow (>\text{M} \;/\; _, C)\} \vdash (\{c_1\} \times \{c_2\}) \rightarrow =M, \quad f_{mcmp}$$

2. and to summarize comparison evaluations regarding multiple properties:

$$\{\{c_1\} \rightarrow (>\text{M} \;/\; _, C),\ \{c_2\} \rightarrow (>\text{M} \;/\; _, C)\} \vdash (\{c_1\} \leftrightarrow \{c_2\}), \quad f_{mcmp}$$

$$f_{mcmp}(f_1, c_1, f_2, c_2) = (1 - |f1 - f2|, c_1 * c_2)$$

whereby f_{mcmp} makes the ratio of positive over total evidence in the conclusion depend on the closeness of the frequency values, and the confidence of the conclusion depend on the confidence of the premises.

Please note that it is also possible to obtain evidence against instance similarity from comparative properties by negating the previous two inference rule conclusions. These rules are useful to find the most similar instances by revising the conclusions, and to find the most significant differences which can then be used to trigger the previous comparative inferences to build an instance-relative description for a new instance to be encoded/described relative to the other.

4 Experiment

The new capability to form comparative relations was also tested in a use case with a Transbot robot. The following is a brief description of the robot hardware and the software architecture used to demonstrate the novel capability. In the end, the experimental setup is described.

Transbot Robot

Transbot is a tracked mobile robot with manipulator arm. Due to its tracks, it is able to perform in outdoor environments, including off-road conditions. It is based on ROS (Robot Operating System) Melodic running on Ubuntu 18.04

LTS and can be programmed using C/C++ and Python. Transbot is useful for building various robotic applications. In terms of hardware, it is equipped with a NVIDIA Jetson Nano board, which acts as the control unit to coordinate and implement the robot's behaviour, and a set of sensors, such as an Astra Pro depth camera and Slamtec RPLIDAR A1 Lidar sensor. Due to its Lidar and depth camera which allows for Simultaneous Localization and Mapping, Transbot is able to map its environment to carry out exploration tasks. Also, it includes robotic arm with three degrees of movement controlled with separate servo motors, which allows the robot to perform various object manipulation tasks (such as object grasping).

The Robot Architecture

The novel way to form comparative relations was integrated in the latest version of ONA (which will be v0.9.2). To show it, a robotic application was realized using the Transbot robot. To endow Transbot with the capability to use NARS as its reasoning system, a related ROS node was realized. The ROS module is a part of the ONA framework, available on Github[1]. The ROS network structure is illustrated in Fig. 1.

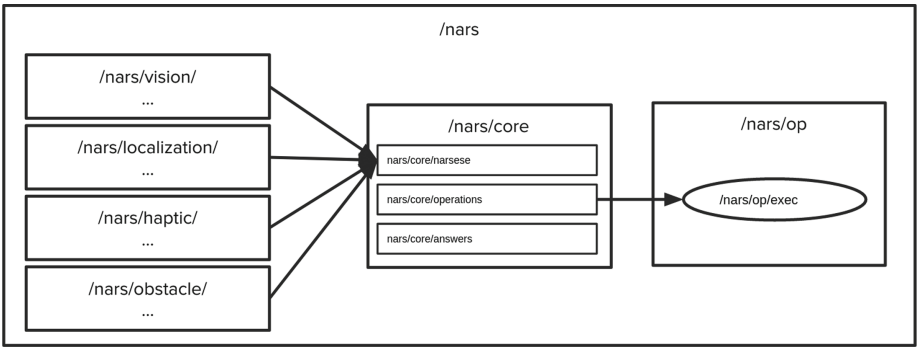

Fig. 1. ROS network for NARS

The network, as illustrated in Fig. 1 consists of channel nodes on input side, a ROS node for NARS (ONA), and a node on the output side for passing on operation calls to ROS components to control the robot.

The channel nodes include (whereby localization and local obstacle detection is omitted as they are not directly required for our experiment):

– vision: a node which applies YOLOv4 [1] on RGB camera input, and merges it with depth camera input to assign a depth to each bounding box. The result is encoded into Narsese, and encodes location, size, and class information in

[1] https://github.com/opennars/OpenNARS-for-Applications.

a way the reasoner can work with. Also color information (red, green, blue) is extracted from within each bounding box detection, by finding the most dominant (in terms of area) color surfaces.
- haptic: responsible for servo feedback of the gripper, to indicate, with a boolean whether the robot managed to get a hold on an object when picking it by monitoring the rate of angle change of the gripper during the relevant part of the pick operation, reporting if it is below a threshold. The feedback is directly encoded as event ($gripper \rightarrow [holding]$) and ($gripper \rightarrow [free]$).

The reasoner core node (as in Fig. 1), directly runs ONA, and accepts new inputs as strings formatted with the Narsese formal language the reasoner works with, by subscribing to the Narsese topic. There is also a topic for real-time Q&A purposes, but this is outside of the scope of this paper. To let ONA control the robot, operation executions are redirected to the Exec node for further dispatch. Hereby the ROS navigation stack is utilized for robot base movements, by publishing *move_base/goal* desired target location goals (both local-relative, and map-absolute) and canceling the previous ones when new operations are invoked by ONA (via *move_base/cancel*). In addition, the manipulator arm of the robot is controlled by directly adjusting the angles of the servo motors based on the location of the visually detected object chosen to be picked and its corresponding depth estimate.

Use Case

Hereby we will make use of that NARS receives both events for color (red, green blue), and for bounding box location and size in the X and Y coordinate for each detected object. This allows the reasoner to build comparative relations as discussed, among the perceived objects, such as relative color and size differences. The task of the robot is to pick the smaller among the two bottles based on the derived comparative relations. As mentioned, in the vision channel, YOLOv4 trained on ImageNet is utilized as an object detection model. The vision channel uses relative location information (relative to the center) together with size information, the output label to form statements of the form $<objectLabel \mapsto [locationComponent]>$ and $<objectLabel \mapsto [sizeComponent]>$ for X and Y dimension respectively. In addition, the most dominant color surface is extracted to give color information to each bounding box corresponding to a detected object, which is $<objectLabel \mapsto [colorComponent]>$ statements for each color component. This encoding makes ONA aware of the detected object types, their colors, and also their position in the camera's field of view as also necessary to pick them up successfully. In terms of mission specification, the following statement specifies picking the smaller bottle among the two observed bottles:

```
<(<{#1 #2} --> bottle> &| <({#1} * {#2}) --> (> sizeY)>) &/
      <({SELF} * {#2}) --> ^pick>) =/> G>.
```

Please note that the most significant differences are also most visible to the system, so even before any goal is given to the system, the system will notice the significant size difference among the two bottles shown in Fig. 2.

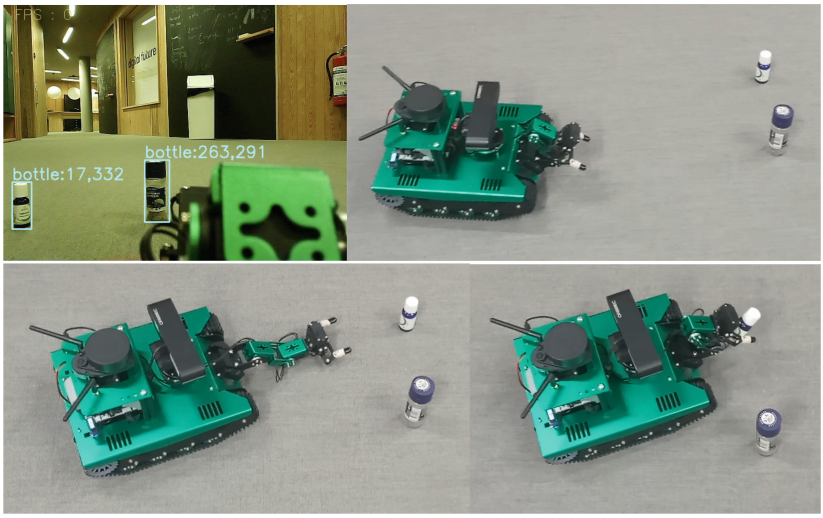

Fig. 2. Selection and picking of smaller bottle

Here, when the two bottles are not in sight, there is a derived goal

```
<{#1 #2} --> bottle>! :|:
```

which leads the system to explore its environment to find said bottles first. The exploration strategies for the robot are outside of the scope of this paper but can be seen in our previous publication [6] and can be combined with established exploration strategies such as based on random trees [12] as long as they can be invoked and stopped at any time by NARS operations.

5 Conclusion

The new comparative reasoning abilities for NARS were presented, including the new relational compound terms and the related inference rules. It was demonstrated that this treatment goes beyond Conceptual Spaces, as combining attributes into pre-defined vectors is not necessary with this approach, and the similarity evaluation is dependent on previously experienced values of the compared attribute in a reference class. Also, it was shown with a concrete experiment how comparative reasoning can lead to cognitive abilities such as being able to compare perceivable quantities of visually detected objects, and to make decisions accordingly on a real robot with a camera. This, as in our example, can be relevant for decision making in intelligent agents, including robots. In the future, experiments where it is required to learn behaviors with comparative relations as a precondition will be shown as well. This ability is a direct consequence of NARS being able to incorporate derived events in procedure learning.

References

1. Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M.: YOLOv4: optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934 (2020)
2. Chella, A.: A cognitive architecture for music perception exploiting conceptual spaces. In: Zenker, F., Gärdenfors, P. (eds.) Applications of Conceptual Spaces. SL, vol. 359, pp. 187–203. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-15021-5_10
3. Chella, A., Dindo, H., Infantino, I.: Imitation learning and anchoring through conceptual spaces. Appl. AI **21**(4–5), 343–359 (2007)
4. Chella, A., Frixione, M., Gaglio, S.: Conceptual spaces for computer vision representations. Artif. Intell. Rev. **16**, 137–152 (2001). https://doi.org/10.1023/A:1011658027344
5. Gardenfors, P.: Conceptual Spaces: The Geometry of Thought. MIT Press, Cambridge (2004)
6. Hammer, P., Isaev, P., Lofthouse, T., Johansson, R.: ONA for autonomous ROS-based robots. In: Goertzel, B., Iklé, M., Potapov, A., Ponomaryov, D. (eds.) AGI 2022. LNCS (LNAI), vol. 13539, pp. 231–242. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-19907-3_22
7. Hammer, P., Lofthouse, T.: 'OpenNARS for applications': architecture and control. In: Goertzel, B., Panov, A.I., Potapov, A., Yampolskiy, R. (eds.) AGI 2020. LNCS (LNAI), vol. 12177, pp. 193–204. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-52152-3_20
8. Latapie, H., Kilic, O., Thórisson, K.R., Wang, P., Hammer, P.: Neurosymbolic systems of perception & cognition: the role of attention. Front. Psychol. 2105 (2022)
9. Lieto, A., Chella, A., Frixione, M.: Conceptual spaces for cognitive architectures: a lingua franca for different levels of representation. Biologically Inspired Cogn. Archit. **19**, 1–9 (2017)
10. Riley, D.A., Ring, K., Thomas, J.: The effect of stimulus comparison on discrimination learning and transposition. J. Comp. Physiol. Psychol. **53**(5), 415 (1960)
11. Santoro, A., et al.: A simple neural network module for relational reasoning. In: Advances in Neural Information Processing Systems, vol. 30 (2017)
12. Umari, H., Mukhopadhyay, S.: Autonomous robotic exploration based on multiple rapidly-exploring randomized trees. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1396–1402. IEEE (2017)
13. Wang, P.: The interpretation of fuzziness. IEEE Trans. Syst. Man Cybern. B Cybern. **26**(4), 321–326 (1996)
14. Wang, P.: Recommendation based on personal preference. In: Zhang, Y., Kandel, A., Lin, T., Yao, Y. (eds.) Computational Web Intelligence: Intelligent Technology for Web Applications, pp. 101–115. World Scientific Publishing Company, Singapore (2004)
15. Wang, P.: Rigid Flexibility: The Logic of Intelligence. Springer, Dordrecht (2006). https://doi.org/10.1007/1-4020-5045-3
16. Wang, P.: Non-Axiomatic Logic: A Model of Intelligent Reasoning. World Scientific, Singapore (2013)
17. Zambaldi, V., et al.: Relational deep reinforcement learning. arXiv preprint arXiv:1806.01830 (2018)