



HAL
open science

From Functional Affordances to Reciprocal Dispositions: An Ontological Representation

Bastien Dussard, Aurélie Clodic, Guillaume Sarthou

► To cite this version:

Bastien Dussard, Aurélie Clodic, Guillaume Sarthou. From Functional Affordances to Reciprocal Dispositions: An Ontological Representation. International Conference on Social Robotics (16th Edition), Oct 2024, Odense, Denmark. hal-04776607v2

HAL Id: hal-04776607

<https://hal.science/hal-04776607v2>

Submitted on 12 Nov 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

From Functional Affordances to Reciprocal Dispositions: An Ontological Representation

Bastien Dussard¹[0009-0001-8800-8853], Aurélie Clodic¹[0009-0009-6484-8143], and
Guillaume Sarthou¹[0000-0002-4438-2763]

LAAS-CNRS, Université de Toulouse, CNRS, Toulouse, France
`firstname.surname@laas.fr`

Abstract. In human-robot interaction, the collaboration of agents stems from what each agent is capable of doing to successfully complete the objective. Thus, for a robot to efficiently interact with its environment, it is crucial to have knowledge over the actions it is capable of performing. This paper presents a step toward the consideration of agentive aspects in the representation of action possibilities for an agent, i.e. affordances. Building upon the dispositional theory, this work aims at matching the dispositions of entities with the capabilities of the agents. Using an ontological representation and inference mechanisms, this contribution aims to autonomously infer affordance relations linking an agent to entities in a reactive way. As affordance relations stem from the match between the dispositions of entities and the capabilities of the agents, we propose a reasoner which allows to link the corresponding entities given each affordance’s requirement. Furthermore, as certain actions may necessitate the combination of multiple entities, this work also addresses the representation and inference of affordances for such combinations. These inferences provide the agent with a self-awareness capability about which actions are feasible with respect to the entities in the environment, while also leveraging changes that may occur in the environment.

Keywords: functional affordances · knowledge representation · ontology.

1 Introduction

For a robot to effectively act on its environment, a key aspect is the knowledge and the understanding of what it can do. Nevertheless, while the set of actions feasible by a given robot can be hard-coded, they do not necessarily correspond to the set of actions the robot can do at a given moment. Indeed, what an agent can perform first depends on its **capabilities**, usually defined as “*the ability to carry out a type of activity*” [20], themselves depending on the agent’s components. However, as these components can evolve over time, either through the use of tools or because of possible failures, the set of capabilities can change over time, and consequently the set of feasible actions as well. While being aware of its action possibilities is key, considering the context of human-robot

interaction, being aware of others’ action possibilities is also mandatory. Indeed, where such introspection can help to act, the estimation regarding the others can help for monitoring purposes as well as estimating how they could help for given tasks. However, where robotic agents have a set of capabilities enabled by their components, human agents can be considered as having a set of common ground capabilities which can still evolve through the use of tools or equipment.

Which actions an agent can perform are also influenced by both its environment and the entities within it, according to their **dispositions**. Indeed, even if a robot is capable of pouring, it cannot perform the action if there is no container to hold the liquid. According to the dispositional theory introduced by Turvey in [24] and refined by Chemero in [4], a disposition is “*the property of a thing that is a potential*”. This theory suggests that each disposition has a complement, which represents the actualizing circumstance of this action possibility. This disposition can be complemented by an agentive entity, which results in the emergence of an affordance. The concept of **affordance** was first introduced by Gibson in [7] to describe what the environment offers to an agent based on its capabilities. Over the years, refinements to this concept have been proposed in the field of ecological psychology [4, 24], industrial design [15], and robotics [16]. One of the major points of disagreement between these formalisms is the nesting background of these affordances. Turvey [24] views affordances as dispositional properties of the environment which can be actualized by an agent’s effectivities. Chemero [4] agrees with the dispositional aspect of affordances but differs by stating that they reside in the agent-environment system and that they are relational properties between an agent and dispositions of the objects in the environment. In this work, we chose to consider affordances as defined by Chemero¹, considering them as the match between an agent’s capability and the complementing disposition of another entity. Hence, a grasping affordance exists between an agent who has the grasping capability and an object which has the disposition to be graspable. This formalism is best suited for our approach as its relational nature facilitates its usage for knowledge representation. Indeed, if we had to translate those views into the ontological world, we could consider Turvey’s as an inheritance link and Chemero’s as a relation between individuals.

While dispositions are at least partially responsible for the affordances conceptualized in both views, the complementarity between dispositions of non-agentive entities can also contribute to the emergence of affordances. The concept of **reciprocal dispositions** refers to entities sharing complementary dispositions (e.g. the knife being dirty has a *Washable* disposition and the dishwasher has a *Washing* disposition). Martin introduced the concept of reciprocity of dispositions in [12], describing it as “*partners for a mutual manifestation that is*

¹ Since our work aims at integrating affordances in a knowledge base, i.e. an internal representation, one could see the contradiction with the anti-representationalist view defended by the ecological psychology field [5]. However, as explained in [29], “*roboticists generally extract features as a basis for affordance detection and learning, thereby implicitly building an internal representation.*”, motivating that the computational perspective over affordances is meaningful for robotics.

their common product". Those dispositional reciprocities can thus be considered as "systems" which lead to the emergence of new affordances.

This work falls within the DACOBOT architecture [19] which is a knowledge-centred robotic architecture. It is based on an ontological description of the robot's knowledge, which includes elements of the environment and some simple robot descriptions. In this paper, we build upon our prior work [6], in which we have proposed a representation of capabilities that allows automatic inferences based on the components an agent owns.

The main contribution of this paper is an **ontological representation of affordances based on agents' capabilities and objects' dispositions**. While most work representing affordances either consider implicit or static representation without the agentive aspect, our representation enables **inference mechanisms** over the agent's affordance relations based on its previously inferred capabilities and the dispositions of the objects in its environment. In this paper, we show that our representation is suitable both for **functional affordances** (affordances from an agent to an entity) and **reciprocal dispositions** (combination of two entities to be used together). In addition, thanks to its integration into our robotic architecture, we show that knowledge about these affordance relations can be maintained and updated over time, with regard to the current state of the world and the current agent's capabilities.

In Sec. 2 we discuss related work and how our contribution addresses new issues. We provide in Sec. 3 a brief ontology formalism and a presentation of the used framework. Our two main patterns are then presented in Sec. 4 and Sec. 5. Finally, we present in Sec. 6 an analysis of the reasoning processes on a robot activity and we conclude this paper in Sec. 7 by discussing possible future work.

2 Related Work

The concept of affordance is essential in robotics as it incorporates the idea of action possibility. Knowledge of how an agent can act on its environment is crucial as it enables effective interaction with its environment and provides efficient task attribution in a collaborative process. However, this concept is often considered implicitly rather than represented in the form of a knowledge representation. Most of the research focused on object recognition [14] to detect affordances, or manipulation [2,13] to learn affordance relations over objects. These image-based methods for detecting affordances suffer from a lack of explainability. However, with the recent advances over the Large Language Model's capabilities, some methods have been proposed to tackle those problems by combining LLMs with more classical approach to overcome the current methods' limits [22]. The aforementioned research makes use of the affordance concept in order to facilitate the execution of robotic tasks. However, they fail to take into account the agentive aspect, as they consider the agent's capabilities to be pre-defined and unchanging.

In order to enhance explainability, affordances can be grounded into a semantic knowledge representation. The use of knowledge bases, such as ontolo-

gies, provides easier extensibility while allowing representation and reasoning about those action possibilities. Nevertheless, in such domain, affordances are often expressed implicitly without conceptualizing the core notion or any reasoning, using either specific classes of inheritance over objects [11] (e.g. *Mug, isA, CanBeManipulated*), relations from an agent over an object [8] (e.g. *bob, canGrab, mug*), or relations from an object over an affordance entity [21] (e.g. *mug, hasAffordance, pourability*).

In order to conceptualize the affordance notion and enhance its applicability, some work did focus on the concept of **functional affordance**. Introduced in [10], this concept aims at representing objects based on their functions, allowing among others for object substitution during task planning. Awaad et al. [1] relate objects to their functions at the class level (e.g. a cup is used for drinking from and for holding) but without representing the object’s parts associated with such affordances. This concept has also been implemented in [26] to enhance a perception system. Relying on a Part Functional Affordance Schema which binds geometrical features and affordances (e.g. sharp edge implies an incision-ability affordance), they defined a set of **Conceptual Equivalence Classes** to represent object concepts given the geometry of their parts and their corresponding affordances. For instance, a knife is defined as the composition of a blade providing an incision-ability affordance and a handle providing a grasp-ability affordance. Thanks to such a representation, they have shown to be able to find in a visual scene objects with equivalent functionalities, leading to the creation of the Affordance Network database (AfNet) [27]. Nevertheless, while considering some reasoning at the instance level, those approaches do not consider the capabilities of a given agent to use the described functions. Consequently, what they call affordances should rather be considered as dispositions of entities.

Although representing the functions of objects is a crucial initial first step, it may not be sufficient to know the action possibilities they provide. This is particularly true for everyday objects, which are often used with other entities. This complementarity between object aspects can be referred to as reciprocal dispositions [12] and is grounded in the dispositional theory of Turvey [24]. For example, with AfRob [28] (an extension of AfNet), the authors introduce **affordance dualities** as the interaction between entities given their complementarity. For instance, a pen having an engrave-ability affordance and a paper sheet having a display-ability affordance can be seen as complementary. However, since those pairs refer to a match between dispositional aspects of non-agentive entities, we consider this conceptualization as reciprocal dispositions and thus not as affordances. More focused on affordance representation, Beßler et al. in [3] propose an ontological formalism that defines an affordance as a description of a disposition. This formalism is built upon the **bearer/trigger** roles, with the bearer being the entity holding the disposition to be actualized by a suitable other entity, the trigger. In such a way, a dirty piece of cutlery would be the bearer of a dishwasher that would be the trigger. Nevertheless, this approach doesn’t leverage the agentive aspect of how the agents can interact with those entities. Indeed, this formalism is not aimed to be instantiated in a given scene

Table 1: Summary of the major features identified for the most representative contributions in the field.

	[8, 11, 21]	[1]	[26–28]	[3]	[25]	[23]	Our
Formalism	-	-	-	Turvey	-	Turvey	Chemero
Reasoning method	-	DL Query	DL query	DL Query	Rule	-	DL logic & Custom reasoner
Functional affordance	No	Yes	Yes	No	Yes	Yes	Yes
Dispositional Match	No	No	Yes	Yes	No	Yes	Yes
Agentive aspect	No	No	No	No	Yes	Yes	Yes
Instantiation	Yes	Yes	No	No	Yes	No	Yes
Dynamic possibility	No	No	No	No	No	No	Yes

but rather to answer competency questions, such as “*What can this be used with?*” or “*What can this be used for?*”. Such questions are thus represented with DL queries based on Generic Ontology Design Patterns that have to be handcrafted for each specific question to be answered. Consequently, this formalism is not suitable for dynamically representing affordances in an ontology and automatic affordance discovery with regard to a given situation.

While the previous contributions focused on the affordance concept formalism, either not considering instantiation or automatic inference, Umbrico et al. in [25] present an affordance rule allowing to infer opportunities for actions. However, while considering the agentive aspects, the inference mechanism relies only on a single rule (one per affordance type) lacking of genericity by embedding every aspect of the affordance. A generic formalization taking into account agentive and object dispositions has been proposed by Toyoshima et al. [23]. In contrast to previous work, the authors adopt Turvey’s perspective, defining the **affordance-effectivity complex** as the sum of reciprocal dispositions between an agent (**effectivity**) and an object (**affordance**). Therefore, the affordance concept is not viewed as a relation between entities but as a disposition. This formalism accounts for both individual (instance) and family (class) directed affordances. Indeed, while the knife concept brings a set of affordances, a specific instance can hold additional ones (e.g. its dirtiness). It allows to leverage both agent capabilities and entity dispositions as the pattern is intended to represent the environment’s perspective. Although formalized as a generic representation of affordances, this formalism requires the creation of multiple individuals to represent a single affordance description. This approach provides a rich and precise representation of affordances, but does not allow for inference over action possibilities. A sum up of the main proposed contribution is presented in Table. 1 with regard to the main features identified.

3 Ontological Framework

An ontology, as defined by Gruber in [9], is a “*formal specification of a shared conceptualization*” which allows to define concepts, individuals, and relationships. One of the strengths of such a knowledge representation is the ability

Table 2: Symbols used to define an ontology

<u>A: ABox entities/indiv</u>	<u>T: TBox classes/concepts</u>
<i>A</i> : set of entities	<i>T</i> : set of classes
<i>C</i> ₀ : entities' direct types	<i>H</i> : classes inheritance links
<i>R</i> : relations between entities	<i>E</i> : equivalence classes
<u>R: RBox roles/properties</u>	
<i>P</i> : set of properties	<i>I</i> : properties inheritance links
<i>Chain</i> : set of chained properties	<i>Inv</i> : properties inverses
<i>Dom</i> : properties domains sets	<i>Ran</i> : properties ranges sets

to infer new knowledge via *Description Logic* (DL) based reasoning. Such reasoning is enabled in several manners, either via a query that answers questions with concepts/entities matching a desired pattern or at the instance level via DL logic rules. Following the ontological formalism presented in [17], the knowledge base K is composed of the Role Box (**Rbox**) defining the properties, the Terminological box (**TBox**) defining the concept hierarchies, and the Assertional box (**ABox**), defining the individuals with their inheritance given the TBox and the relations between individuals via the properties given by the RBox. Table. 2 illustrates a sum up of the ontological symbols and basic reasoning mechanisms used in this work are presented below.

Defining an equivalence between classes allows to infer inheritance relations between individuals which share common conceptualizations. An equivalence class $(t, e) \in E$ is defined as the link between a class $t \in T$ and a class expression containing either another class from T or a complex class expression e . Formally, if $(t, e) \in E, \forall a \in A | e(a) \subset K$, then $(a, t) \in C_0$. As an example, an equivalence to the class *Father* can be *Father* \equiv *Male and (hasChild some Human)*.

Expressing a property as a path between several properties can allow to deduce new knowledge between individuals connected together. A property chain allows to link individuals together given a list of properties. Given the chain $p_1 \bullet p_2 \Rightarrow p_3$, if $(x, p_1, y), (y, p_2, z) \subset R$, then $(x, p_3, z) \in R$. e.g. *hasParent* \bullet *hasMother* \Rightarrow *hasGrandMother*.

The attributes *Domain* and *Range* of a property aim at specifying the classes linkable together. Usually used for consistency checking, they can also be used to infer the class of individuals involved in the relation. Given the property p and the individuals $(x, y) \subset A$, if $(x, p, y) \in R$ then $\forall t \in Dom(p), (x, t) \in C_0$ and $\forall t \in Ran(p), (y, t) \in C_0$. For example, if the property *ownsDog* has *Human* as Domain and *Dog* as Range, and the relation $(bob, ownsDog, medor)$ is asserted, then the inheritance links between the *bob* individual and the *Human* class, as well as between the *medor* individual and *Dog* class, are inferred.

To manage the ontology and reason over it, we use the software *Ontologienus* [18] which has been specifically designed for dynamic reasoning. In the context of this work, we will use its built-in reasoners for inverse axioms, chained

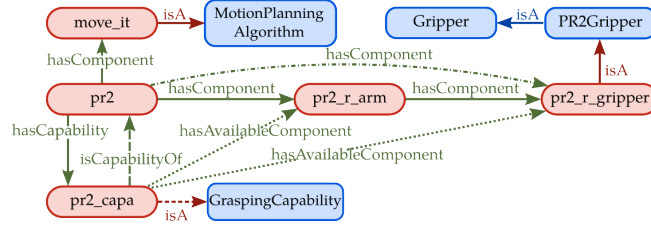


Fig. 1: **Ontological pattern example allowing the inference of the Grasping capability.** \rightarrow are inferences by inverse, \dashrightarrow are inferences by equivalence, \dashrightarrow are inferences by transitivity, and $\cdots\rightarrow$ are inferences by property chain.

axioms, transitive axioms, and equivalent classes. We will later introduce a custom reasoning process for the purpose of this contribution.

4 Functional Affordances

The concept of functional affordance refers to the intended use of an object according to common ground knowledge. Functional affordances stem from an entity’s dispositions (e.g. a knife has a cutting disposition), but can also be influenced by external factors such as a change in its status (e.g. the knife becoming dirty after usage has a cleanable disposition). In the literature, some ontological work tackled this concept but without taking into account the agent’s capabilities. For instance, a tool having a grip part requires an agent to have the grasping capability in order to actually afford the function provided by the tool. As functional affordances emerge from the capabilities of agents, their representation must leverage such an aspect to provide the set of actual action possibilities.

4.1 Agent Capabilities Representation

Since the affordance relations available to an agent stem from its capabilities, we build upon our previous work proposed in [6]. There, we introduced an ontological pattern to represent and dynamically infer agents’ capabilities based on their components. We represented an agent’s capabilities with a single individual and described each capability by an equivalence class. It allowed to infer inheritance links between the capability individual and each enabled *Capability* class given its components and its already inferred capabilities. An example of this process is shown in Fig. 1 with **GraspingCapability** \equiv **hasAvailableComponent some (Gripper and MotionPlanningAlgorithm)**, resulting in the inference of the relation (*pr2_capa*, *isA*, *GraspingCapability*).

4.2 Object Dispositions Representation

In addition to an agent’s capabilities, enabling an affordance requires certain properties from non-agentive entities. Such properties are referred to as dispositions and can be represented with varying degrees of semantic granularity. In the

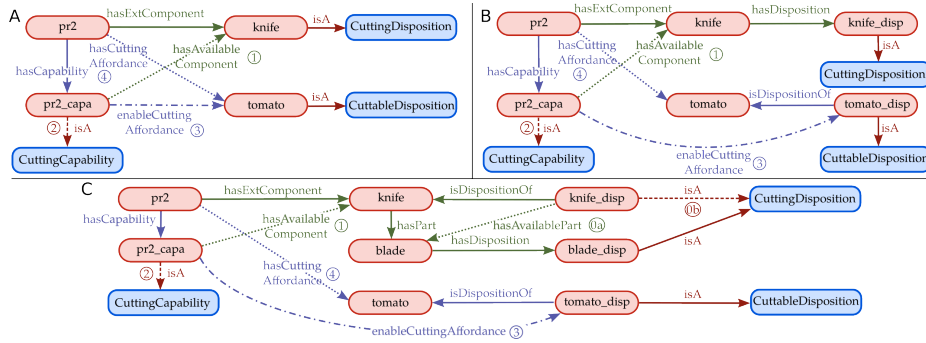


Fig. 2: **Ontological patterns example allowing the inference of the *hasCuttingAffordance* relation** (A) Example with individuals directly inheriting from the corresponding Disposition class. (B) Example with the individuals related to their disposition individuals, which inherits from the corresponding Disposition class. (C) Example with the tool individual having a sub-part. Both have a disposition individual, the tool’s one being inferred by equivalence as the class of its sub-part’s corresponding Disposition class. \rightarrow are inferences by inverse, \dashrightarrow are inferences by equivalence, \dashrightarrow are inferences by property chain and \dashrightarrow are inferences by the custom reasoner. Numbers like \textcircled{x} correspond to the order of inferences.

straightforward case, entities can be represented as directly inheriting from their corresponding dispositions as in Fig. 2.A (e.g. $(knife, isA, CuttingDisposition)$). However, such a representation can lack semantic precision as an entity holds a disposition instead of being one. As dispositions are similar to capabilities but for non-agentive entities, one could represent an entity’s dispositions with a dedicated individual, similarly to the agent capability individual as in Fig. 2.B (e.g. $(knife, hasDisposition, knife_disp)$ and $(knife_disp, isA, CuttingDisposition)$). Furthermore, as sub-parts composing entities can provide dispositions to the whole entity, we can describe them with a pattern similar to our previous work. For example, we can represent a knife as having a blade, which holds the cutting disposition as in Fig. 2.C (e.g. $(knife, hasPart, blade)$, $(blade, hasDisposition, blade_disp)$ and $(blade_disp, isA, CuttingDisposition)$). Given this representation, the property chain $isDispositionOf \bullet hasPart \implies hasAvailablePart$ allows to infer the relation $(knife_disp, hasAvailablePart, blade)$ $\textcircled{0a}$. Once this relation has been inferred, the dispositions of the knife individual can be deduced via the equivalence class $CuttingDisposition \equiv (hasAvailablePart \text{ some } (hasDisposition \text{ some } (CuttingDisposition)))$ (e.g. $(knife_disp, isA, CuttingDisposition)$) $\textcircled{0b}$. Thanks to this inferred relation, the case becomes equivalent to the Fig. 2.B case, in which we asserted the inheritance relation directly. To illustrate a context in which the robot holds a knife, we create the relation $(pr2, hasExtComponent, knife)$. This allows us to use the same property chain as for the agent’s components since $(hasExtComponent, hasComponent) \in I$

and **isCapabilityOf•hasComponent** \implies **hasAvailableComponent**. Thus, the relation $(pr2_capa, hasAvailableComponent, knife)$ is inferred ①. Similarly to an agent acquiring a new component and gaining new capabilities, an agent holding a tool can also gain new capabilities based on the tool’s dispositions. Therefore, by defining CuttingCapability as **CuttingCapability** \equiv **hasAvailableComponent some CuttingDisposition** for Fig. 2.A, the relation $(pr2_capa, isA, CuttingCapability)$ is inferred as $pr2_capa$ validates the class expression ②. The same mechanism can be used for the examples with dedicated disposition individuals by modifying the equivalence class into **CuttingCapability** \equiv **hasAvailableComponent some (hasDisposition some (CuttingDisposition))** for Fig. 2.B/C.

4.3 Custom Reasoner

Having represented the agents’ capabilities and the entities’ dispositions, the next necessary step is to link them into the creation of functional affordances. Given the relational aspect of this concept, we chose to represent an affordance as a property relating an agent to the involved entity (e.g. $(agent, hasXAffordance, entity)$). Hence, we propose a pattern that enables the dynamic inference of functional affordances using standard ontological mechanisms and a custom reasoner. This latter is introduced as standard mechanisms only allow inference of relations between individuals if a relational path between them already exists. In the case of two individuals inheriting from specific classes, no such relational path exists, and thus inferring a link between them with standard means is not feasible. Such a process is necessary as we want to link an agent with an entity if its capabilities (the inheritance links toward Capability classes) match the corresponding dispositions of the entity (the inheritance links toward Disposition classes). The custom reasoner’s role is inspired from the *Dispositional Match* approach of [3], but differs by leveraging the agentive aspect of affordances. For this purpose, we chose to use the Domain and Range attributes of each affordance property to represent the required capabilities and dispositions for a match. The process of this custom reasoner is described as below.

Given the top property *enableAffordance* expressed as $affp$, the set of applicable affordance properties P_{aff} as:

$$P_{aff} = \{p \mid (p, affp) \in I \wedge Dom(p) \neq \emptyset \wedge Ran(p) \neq \emptyset\}$$

Next, for each affordance property, we search which individuals in the ontology match the Domain classes and which match the Range classes attributes. This searching process results in the creation of two sets of candidate individuals, A_D and A_R .

$$A_D = \{a_D \mid \forall t_D \in Dom(p) \wedge (a_D, t_D) \in C\}$$

$$A_R = \{a_R \mid \forall t_R \in Ran(p) \wedge (a_R, t_R) \in C\}$$

Finally, we can link each individual from the candidate sets (A_D, A_R) with each other via the affordance property p .

$$\forall a_D \in A_D, \forall a_R \in A_R \implies (a_D, p, a_R) \in R$$

4.4 Functional Affordance: an Example

According to the examples in Fig. 2 which illustrate the functional affordance of cutting, we define the property *enableCuttingAffordance* with *CuttingCapability* as Domain and *CutableDisposition* as Range. Thanks to the reasoner, the relations $(pr2_capa, enableCuttingAffordance, tomato)$ in Fig. 2.A and $(pr2_capa, enableCuttingAffordance, tomato_disp)$ in Fig. 2B/C are inferred as these individuals do match the required classes ③.

These inferred relations link the agent’s capability individual to the corresponding entity (or entity disposition) individual. As we want to represent the affordance relation between the agent and the entity directly, an extra step is required to infer the relation (e.g. $(agent, hasXAffordance, entity)$). This final step is done by the use of a property chain of the form **hasCapability•enableCuttingAffordance** \implies **hasCuttingAffordance** for Fig. 2.A. As the other examples use an intermediary dispositional individual, the property chain is defined as **hasCapability•enableCuttingAffordance•isDispositionOf** \implies **hasCuttingAffordance**. We are therefore able to infer the relation $(pr2, hasCuttingAffordance, tomato)$ which states that the pr2 robot has a cutting functional affordance toward the tomato ④.

5 Affordances and Reciprocal Dispositions

Until now, we represented affordances as relations between an agent and an entity, respectively through their capabilities and dispositions. However, some non-agentive entities can interact together to provide new action possibilities (affordances) to the agent if they have reciprocal dispositions. Thus, we consider that a pair of reciprocal dispositions creates a Dispositional Match (DM) and that an agent can have affordances toward DMs if it has the right set of capabilities to enable this affordance. To represent such affordances, we expand the pattern described in the previous section, taking into account the specific challenge of creating a new individual.

5.1 Custom reasoner: an Extension

For this purpose, the custom reasoner needs to be extended to enable such additional features to be leveraged in the inference of affordance relations. First, similarly to the way we previously linked a capability to a disposition, we have to link two reciprocal dispositions together ①. Therefore, the process follows the same pattern as for the *enableAffordance* property but with the *enableDM* property linking two dispositions together. Thus, only the first equation is affected by such a change. Given the top property *enableDM* expressed as *dispm*, the set of applicable dispositional properties P_{disp} as:

$$P_{disp} = \{p \mid (p, dispm) \in I \wedge Dom(p) \neq \emptyset \wedge Ran(p) \neq \emptyset\}$$

As illustrated in Fig.3, considering a dirty knife having a *WashableDisposition*, as the dishwasher individual has a reciprocal *WashingDisposition*,

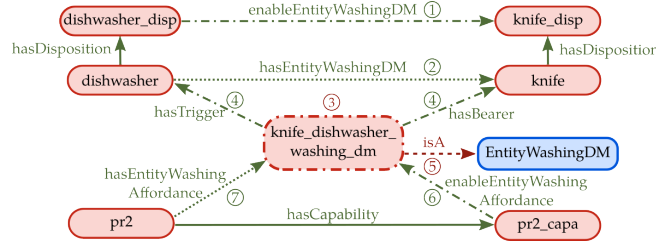


Fig. 3: **Ontological pattern example allowing the inference of a Dispositional Match (DM) and a related affordance.** $-\!-\!-\!>$ are inferences by equivalence, $-\!-\!>$ are inferences by the custom reasoner, and $\cdots\!-\!>$ are inferences by property chain. Numbers like \textcircled{x} correspond to the order of inferences.

we infer that their dispositions can interact together ($(dishwasher_disp, enableWashingDM, knife_disp)$). The transition from the dispositions individuals to the effective entities is once again done through a property chain **hasDisposition•enableWashingDM•isDispositionOf** \implies **hasWashingDM**. We thus infer the relation $((dishwasher, hasWashingDM, knife))$ $\textcircled{2}$.

Once a dispositional match is found, an individual has to be created $\textcircled{3}$ to represent the generated new “system”, to borrow Turvey’s words. This latter is then linked to its bearer and its trigger $\textcircled{4}$ which correspond respectively to the entity holding the actualizable and actualizing dispositions. This process is managed by our custom reasoner and can be expressed as:

$$\begin{aligned} \forall (a_D, a_R) | (a_D, p, a_R) \in R \wedge (p, hasDM) \in I \\ A \leftarrow A \cup \{a_{DR}\}, \\ R \leftarrow R \cup \{(a_{DR}, hasTrigger, a_D), (a_{DR}, hasBearer, a_R)\} \end{aligned}$$

5.2 Dispositional Match: an Example

For this example, our custom reasoner creates a new individual $k_d_washing_DM$ individual. This DM individual is then linked to its bearer/trigger individuals with the relations $(k_d_washing_dm, hasBearer, knife)$ and $(k_d_washing_dm, hasTrigger, dishwasher)$.

As this DM individual represents a sum of reciprocal dispositions, it is by essence a disposition and has to be represented in consequence. To do so, we use an equivalence class which at least includes the involved dispositions. Because our goal is to obtain a representation of the affordances applicable for an agent, the equivalence class must leverage the dispositions required to use the affordance. For our example, the match between *Washing* and *Washable* dispositions is not sufficient in order to get those two entities to interact together. Thus, we add to the representation that the washable individual must have a *GraspableDisposition*. The definition of the equivalence class **EntityWashingDM** \equiv (**hasTrigger some (WashableDisposition and GraspableDisposition)** and **hasBearer some (WashingDisposition)**) results in the

inference of the relation $(k_d_washing_dm, isA, EntityWashingDM)$ ⑤. In a similar way, the disposition of the dishwasher’s button to be pressed could be inserted in such an equivalence.

At this stage, the DMs are expressed through their dispositions and the affordances can be inferred based on those using the functional affordance mechanism presented earlier. Similarly to the case of Fig. 2.A, the agent’s capabilities can thus enable the affordance ⑥ which can then be applied to the agent itself ⑦.

6 Results

This section describes the reasoning process happening in our framework for the examples used throughout this paper. It aims at demonstrating how these patterns, which represent affordances, evolve with the modifications occurring in the environment and can help the representation of the robot’s action possibilities for a given situation.

To illustrate this process, we will consider a scenario with a robot helper in a kitchen environment. Let us consider a “making a tomato salad” task, with the robot helper having access to the following equipment: a knife, a tomato, and a dishwasher. We assume that the robot has two grippers, a motion planning algorithm, and an object localization node. It would therefore be inferred that it is capable of grasping and localizing entities.

Based on its grippers which provided the grasping capability, the robot initially has a grasping affordance toward the knife and the tomato due to their *Graspable* dispositions. Furthermore, since the knife and the tomato have reciprocal dispositions due to their *Cutting* and *Cuttable* dispositions, a DM individual would be created to represent their interacting-ability. Using the dispositional match algorithm, the robot would be inferred as having an entity cutting affordance toward this DM individual as it has the right set of capabilities.

When the robot grasps the knife, its capabilities would be updated to include the knife’s cutting disposition as a cutting capability of its own. Therefore, the robot would have a “direct” affordance toward the tomato, meaning it currently affords this action but also an “indirect” affordance toward the DM individual between the tomato and the knife. This latter would thus represent an action possibility requiring another action to be executed before in order to actualize the disposition.

Moreover, assuming the robot has sliced the tomato, the knife’s status would change to dirty and thus updating its dispositions with a *Washable* disposition. This, in turn, would create another DM individual due to the dishwasher’s reciprocal disposition of *Washing*. The robot helper, holding the necessary capabilities for this affordance, would be inferred as having an affordance relation with this new DM individual, meaning that it can wash the knife with the dishwasher.

Furthermore, one could argue that for washing the knife with the dishwasher, a washing tablet is required. This latter and the dishwasher would have reciprocal dispositions, leading to the creation of a new DM individual. The knife and this DM individual would, in turn, lead to the creation of another DM individual

representing the washing affordance, with the knife being the bearer but the dishwasher-washing tablet DM as the trigger.

7 Conclusion and Future Work

In this paper, we have proposed an ontological representation of the functional affordance concept based on agentive aspect and entity dispositions. Unlike the existing literature, our representation of the affordance set is dynamic and maintainable given the current state of the environment. These patterns leverage affordances involving multiple entities by building on the concept of reciprocal dispositions, while also taking into account the agent’s capabilities.

We are aware that our custom reasoner is not perfectly aligned with the standard inference mechanisms but is required to be able to dynamically infer affordances. Nevertheless, we believe that the proposed patterns are still useful and meaningful for describing affordances and dispositional matches in a static way, as this is the most common use.

Concerning future improvements, we could include more properties to represent the actual affordance emergence, such as physical constraints, for example the maximum weight a robot can lift, but also take into account the affordance emerging between agents.

Acknowledgments. This work has been supported by the Effective Learning of Social Affordances (ELSA) project ANR-21-CE33-0019.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Awaad, I., Kraetzschmar, G.K., Hertzberg, J.: The role of functional affordances in socializing robots. *International Journal of Social Robotics* (2015)
2. Barck-Holst, C., Ralph, M., Holmar, F., Kragic, D.: Learning grasping affordance using probabilistic and ontological approaches. In: *International Conference on Advanced Robotics*. IEEE (2009)
3. Bekler, D., Porzel, R., Pomarlan, M., Beetz, M., Malaka, R., Bateman, J.: A formal model of affordances for flexible robotic task execution. In: *ECAI*. IOS Press (2020)
4. Chemero, A.: An outline of a theory of affordances. *Ecological Psychology* (2003)
5. Chemero, A., Turvey, M.T.: Gibsonian affordances for roboticists. *Adaptive Behavior* (2007)
6. Dussard, B., Sarthou, G., Clodic, A.: Ontological component-based description of robot capabilities. In: *International Workshop on Working towards Ontology-based Standards for Robotics and Automation* (2023)
7. Gibson, J.J.: *The ecological approach to visual perception* (1979)
8. Gonçalves, P.J., Torres, P.M.: Knowledge representation applied to robotic orthopedic surgery. *Robotics and Computer-Integrated Manufacturing* (2015)
9. Gruber, T.R.: A translation approach to portable ontology specifications. *Knowledge acquisition* (1993)

10. Hartson, R.: Cognitive, physical, sensory, and functional affordances in interaction design. *Behaviour & information technology* (2003)
11. Lemaignan, S., Warnier, M., Sisbot, E.A., Clodic, A., Alami, R.: Artificial cognition for social human–robot interaction: An implementation. *Artificial Intelligence* (2017)
12. Martin, C.B.: *The mind in nature*. OUP Oxford (2010)
13. Montesano, L., Lopes, M., Bernardino, A., Santos-Victor, J.: Learning object affordances: from sensory–motor coordination to imitation. *IEEE Transactions on Robotics* (2008)
14. Myers, A., Teo, C.L., Fermüller, C., Aloimonos, Y.: Affordance detection of tool parts from geometric features. In: *International Conference on Robotics and Automation (ICRA)*. IEEE (2015)
15. Norman, D.A.: *The design of everyday things* (2002)
16. Şahin, E., Cakmak, M., Doğar, M.R., Uğur, E., Üçoluk, G.: To afford or not to afford: A new formalization of affordances toward affordance-based robot control. *Adaptive Behavior* (2007)
17. Sarthou, G.: Knowledge representation and exploitation for interactive and cognitive robots. Ph.D. thesis, Université Paul Sabatier - Toulouse III (2021)
18. Sarthou, G., Clodic, A., Alami, R.: Ontologenius: A long-term semantic memory for robotic agents. In: *International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE (2019)
19. Sarthou, G., Mayima, A., Buisan, G., Belhassen, K., Clodic, A.: The director task: a psychology-inspired task to assess cognitive and interactive robot architectures. In: *International Conference on Robot & Human Interactive Communication (RO-MAN)*. IEEE (2021)
20. Solano, L., Romero, F., Rosado, P.: An ontology for integrated machining and inspection process planning focusing on resource capabilities. *International Journal of Computer Integrated Manufacturing* (2016)
21. Su, Y.F., Liu, A., Lu, W.H.: Improving robot grasping plans with affordance. In: *International Conference on Advanced Robotics and Intelligent Systems (ARIS)*. IEEE (2017)
22. Tang, J., Zheng, G., Yu, J., Yang, S.: Cotdet: Affordance knowledge prompting for task driven object detection. In: *IEEE/CVF International Conference on Computer Vision* (2023)
23. Toyoshima, F., Barton, A.: A formal representation of affordances as reciprocal dispositions. *TriCoLore (C3GI/ISD/SCORE)* (2018)
24. Turvey, M.T.: *Ecological foundations of cognition: Invariants of perception and action*. American Psychological Association (1992)
25. Umbrico, A., Cortellessa, G., Orlandini, A., Cesta, A.: Modeling affordances and functioning for personalized robotic assistance. In: *KR* (2020)
26. Varadarajan, K., Vincze, M.: Ontological knowledge management framework for grasping and manipulation. In: *IROS Workshop: Knowledge Representation for Autonomous Robots* (2011)
27. Varadarajan, K.M., Vincze, M.: Afnet: The affordance network. In: *Asian conference on computer vision*. Springer (2012)
28. Varadarajan, K.M., Vincze, M.: Afrob: The affordance network ontology for robots. In: *IEEE/RSJ international conference on intelligent robots and systems* (2012)
29. Zech, P., Haller, S., Lakani, S.R., Ridge, B., Ugur, E., Piater, J.: Computational models of affordance in robotics: a taxonomy and systematic classification. *Adaptive Behavior* (2017)