



HAL
open science

Purpose for Open-Ended Learning Robots: A Computational Taxonomy, Definition, and Operationalisation

Gianluca Baldassarre, Richard Duro, Emilio Cartoni, Mehdi Khamassi, Alejandro Romero, Vieri Giuliano Santucci

► To cite this version:

Gianluca Baldassarre, Richard Duro, Emilio Cartoni, Mehdi Khamassi, Alejandro Romero, et al.. Purpose for Open-Ended Learning Robots: A Computational Taxonomy, Definition, and Operationalisation. 2024. hal-04670576

HAL Id: hal-04670576

<https://hal.science/hal-04670576v1>

Preprint submitted on 12 Aug 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.

Purpose for Open-Ended Learning Robots: A Computational Taxonomy, Definition, and Operationalisation

Gianluca Baldassarre^{*†}, Richard J. Duro^{*‡}, Emilio Cartoni[†], Mehdi Khamassi[§],
Alejandro Romero[‡], Vieri Giuliano Santucci[†]

**These two authors contributed equally to this work*

[†]Institute of Cognitive Sciences and Technologies, National Research Council of Italy
{gianluca.baldassarre, emilio.cartoni, vieri.santucci}@istc.cnr.it

[‡]Integrated Group for Engineering Research, CITIC, Universidade da Coruña
{richard.duro, alejandro.romero.montero}@udc.es

[§]Institute of Intelligent Systems and Robotics, Sorbonne University / CNRS, Paris, F-75005, France
mehdi.khamassi@sorbonne-universite.fr

Abstract—¹ Autonomous open-ended learning (OEL) robots are able to cumulatively acquire new skills and knowledge through direct interaction with the environment, for example relying on the guidance of intrinsic motivations and self-generated goals. OEL robots have a high relevance for applications as they can use the autonomously acquired knowledge to accomplish tasks relevant for their human users. OEL robots, however, encounter an important limitation: this may lead to the acquisition of knowledge that is not so much relevant to accomplish the users' tasks. This work analyses a possible solution to this problem that pivots on the novel concept of 'purpose'. Purposes indicate what the designers and/or users want from the robot. The robot should use internal representations of purposes, called here 'desires', to focus its open-ended exploration towards the acquisition of knowledge relevant to accomplish them. This work contributes to develop a computational framework on purpose in two ways. First, it formalises a framework on purpose based on a three-level motivational hierarchy involving: (a) the purposes; (b) the desires, which are domain independent; (c) specific domain dependent state-goals. Second, the work highlights key challenges highlighted by the framework such as: the 'purpose-desire alignment problem', the 'purpose-goal grounding problem', and the 'arbitration between desires'. Overall, the approach enables OEL robots to learn in an autonomous way but also to focus on acquiring goals and skills that meet the purposes of the designers and users.

Keywords: Purpose, desires, goal, open-ended learning, intrinsic motivations, reinforcement learning, formalisation, alignment, grounding.

I. INTRODUCTION

The aim of this work is to illustrate and formalise a computational framework on the concept of 'purpose', a novel

means to enhance the utility of open-ended learning (OEL) robots and at the same time bias and constrain their autonomy. Autonomous OEL robots are able to cumulatively acquire new skills and knowledge through a direct interaction with the environment, for example by relying on the guidance of intrinsic motivations (e.g., novelty, surprise, and the acquisition of competence) and the self-generation of goals leading to skill acquisition [1]–[3]. OEL robots have a great application potential as they can use the autonomously acquired skills and knowledge to accomplish tasks relevant for human users [4], [5].

Robot autonomy is an essential component to ensure versatility and adaptability. However, an important open problem of OEL robots is that they get hooked on any possible experience deemed interesting. For example, the robots' intrinsic motivations can push them to further explore any action that produces surprising effects. While this is useful to reduce uncertainty about actions that are relevant for users' tasks, this also leads robots to invest their time and learning resources on the multitude of possible experiences offered by the environment, which is not necessarily useful. As a consequence, OEL robots can possibly acquire shallow knowledge and skills that are not really relevant to accomplish the users' tasks. In addition, OEL could lead to misalignment of the robot with human values if it pursues goals and performs actions contrary to them.

This work analyses a possible solution to these problems that pivots on the novel concept of *purpose*. A purpose indicates what the designer and/or user want from the robot in a representation that is particular to them. For example, the designer might want that, whatever the use of the robot may be, it does not hurt people or damage things. The user might want the robot to accomplish specific goals. Another user might want the robot to autonomously acquire all possible knowledge and skills in relation to a certain class of goals, for example involving the manipulation of fruits and vegetables, so that it is able to readily solve tasks drawn from that class when required.

¹This work has received funding from: the European Union's Horizon 2020 Research and Innovation Programme, GA No 101070381, project 'PILLAR-Robots - Purposeful Intrinsically motivated Lifelong Learning Autonomous Robots'; the 'European Union, NextGenerationEU, PNRR', project 'EBRAINS-Italy - European Brain ReseArch INfrastructureS Italy', MUR code IR0000011, CUP B51E22000150006 and project 'FAIR - Future Artificial Intelligence Research', MUR code PE0000013, CUP B53C22003630006; and the European Innovation Council, GA No 101071178, project 'Counterfactual Assessment and Valuation for Awareness Architecture'.

The general idea is that the robot should use the purpose to focus its open-ended exploration towards the acquisition of knowledge that is indeed relevant for the designer/user purpose. In order to do so, the robot should somehow encode the user’s purpose into an inner motivational representation here called *desire*. An important feature of the purpose framework is that desires are domain-independent. In fact, to allow the robot to accomplish the purpose in different, probably previously unknown, domains, this internal motivational representation should be domain independent. Then, once in a particular domain, domain dependent goals can be discovered that cater for the purpose. This can be done by linking the robot’s desires, corresponding to the users’ purposes, to specific domain-dependent *goals* (e.g., the purpose and desire of ‘sorting fruits in different containers’ might involve different goals in different domains involving specific fruits, vegetables, containers, and contexts).

The purpose concept could also permit avoiding behaviours that do not meet human values, rules and conventions (e.g. ‘do not hurt humans’, ‘do not break things’, ‘do not interrupt humans while they are talking together’). While the AI value alignment problem [6], [7] is beyond the scope of this article, we will discuss how the purpose concept could be further investigated in this direction.

This work contributes to the definition of a computational framework aiding the design of OEL robots enhanced by purpose. In this line, Section II introduces the new concept of purpose to constrain open-ended learning and make it useful for technological applications. Section III presents the concept of purpose and related concepts (e.g., desires and goals) in a narrative form. Section IV presents the formalisation of purpose and related concepts. Section VII introduces the key problems that should be addressed to actually employ the concept of purpose within robot architectures. Section VI presents an illustrative scenario in order to clarify the consequences of structuring motivation this way. Finally, section VIII draws conclusions.

II. OPEN-ENDED LEARNING

A. Definition and related concepts

In the fields of robotics and machine learning the concept of OEL refers to the ability of a system to continuously acquire new knowledge and skills without having a specific assigned task [2], [4], [8]. Traditionally machine learning aims to build systems that are trained on a fixed set of data/tasks and are able to generalise on similar new data/tasks. In contrast, OEL involves a continuous autonomous exploration of new data or environments with the aim to progressively improve the acquired knowledge and skills.

One related concept is the one of *lifelong learning* [9]. Both open-ended and lifelong learning refer to the idea that a system should be able to continuously learn and improve over time. However, research under the heading of lifelong learning often focuses on the possibility to be able to continuously modify acquired knowledge and skills, and using what has been learnt previously as scaffolding or transfer to acquire new skills without generating catastrophic interference or forgetting.

Continual learning [10] is another related concept, which is similar to both OEL and lifelong learning. Continual learning refers to the ability of a system to learn from a stream of data over time, rather than from a fixed dataset. However, continual learning typically assumes that the data is sequential and correlated, while OEL is more general and open to exploring a wide range of data sources.

Finally, curriculum learning [11] is another concept related to open-ended learning, but with some important differences. Curriculum learning involves presenting a system with a sequence of tasks or challenges, each building on the previous ones in difficulty, with the goal of gradually constructing the system’s capabilities. While open-ended learning involves exploring a wide range of tasks and environments, curriculum learning is more focused and goal-oriented, with a predefined sequence of tasks (although it has also been studied as an autonomous process).

Open-ended learning can be conceptualised as a form of reinforcement learning where the objective function is not to maximise rewards, but rather to maximise the acquisition of knowledge and skills [4]. This can be formalised if one assume that there are two distinct phases where the robot acts (in reality these phases are commonly mixed). In a first ‘intrinsic phase’, the robot autonomously explores the environment. In a second ‘extrinsic phase’ the robot solves some tasks ‘randomly drawn’ from the same environment and representative of all possible tasks that a user might select. The idea is that during the intrinsic phase the robot should learn as much knowledge and skills as possible so that it can have the maximum performance in solving the tasks of the extrinsic phase. Critically, since the extrinsic-phase goals represent all possible tasks the robot might be called to accomplish in the given environment, the robot’s average performance on them represents an objective measure of the quality of the knowledge that it was able to autonomously acquire during the intrinsic phase. The objective function of the robot can thus be formalised in terms of the performance on the goals of the extrinsic phase:

$$\theta^* = \arg \max_{\theta} E_{g \sim \tau(g)} (E_{\pi(a|s,g,\theta)} R(g)) \quad (1)$$

where θ are the parameters of the robot controller to be optimised, $g \sim \tau(g)$ are the goals drawn from the distribution of all possible goals in the given environment, $R(g)$ is the total reward measuring the performance on goal g , $\pi(a|s, g, \theta)$ is the robot goal-conditioned action policy selecting action a in response to state s and the currently pursued goal g . Finally, θ^* are the optimal parameters that the robot should *search for during the intrinsic phase*, and that would ideally allow the robot to solve any possible goal in the environment.

To effectively learn in an autonomous fashion without any guidance (e.g. during ‘intrinsic phases’), robots should be endowed with motivational mechanisms that leverage the epistemic acquisition of knowledge to drive them to explore the environment, acquire knowledge on it (e.g., world models), and learn policies that reliably accomplish goals. Intrinsic motivations are the most important mechanisms devised to this end [8]. Three major classes of intrinsic motivations

have been highlighted in the literature [1], [12]–[14]. *Novelty-based* intrinsic motivations are mechanisms that generate a learning signal based on the novelty of observations with respect to those previously experienced by the agent and stored in its *memory*. *Surprise-based* intrinsic motivations generate a learning signal based on the amount of *prediction error* of a world model. Finally, *competence-based* intrinsic motivations are mechanisms that generate learning signals on the basis of the *improvement of competence* that the robot gains in accomplishing a given goal.

B. Using IMOL to solve user-assigned tasks: limitations

Intrinsic motivations are an important means to enable robots to learn in an open-ended autonomous way. However, in most cases OEL robots have been trained in simplified laboratory environments, where the possibilities of interaction with the world, and hence the number of things the robot can learn, are limited. In such simplified scenarios, there is a high probability that the acquired skills of the robot will match the subsequent user requests [5]. However, in applications involving real-world complex and unstructured scenarios the amount of new knowledge and skills that the robot can acquire are potentially unbounded. This poses the risk that the robot, guided solely by intrinsic motivations, will focus on exploring a limited portion of space and hence it will remain unable to solve the tasks later assigned to it by the user. It also poses the risk that the robot explores locations, objects or actions that are undesirable or even unacceptable from the point of view of the user or the designer. For example, the robot might search inside a personal bag, break fragile objects while manipulating them, or enter a forbidden room (‘alignment problem’ [6]). To address these issues, in this work we present a three level motivational framework that we call the Purpose framework. In the following section we provide a qualitative overview of its structure and operation.

III. QUALITATIVE OVERVIEW OF THE PURPOSE FRAMEWORK

The purpose framework involves three levels: the designer/user level, the robot level, and the domain level (see Figure 1). The designer/user’s level and the robot level are each formed by two sub-levels. The designer/user level and the domain level are external to the robot. In the following, the term ‘objective’ is used in the context of the three levels in a neutral fashion: to facilitate the description, specific terms are used to refer to objectives within the specific levels and sub-levels as they have specific features.

The first level involves the designer/user. This level contains two sub-levels: the purpose sub-level and human-goal sub-level. The first sub-level is related to the humans’ representations of objectives, called here *purposes*, that they want the robot to accomplish in the environment. An example of purpose is for example ‘having some fruits sorted into different containers’.

The second designer/user sub-level involves different *user-goals*. These user-goals are specific observation representations acquired by the users corresponding to the purpose in

different specific domains. The user-goals are in particular internal representations of desired domain states (third level). For example, in one domain the user-goal might be ‘bananas in a basket and pineapples in a case’; in another domain the user-goal might be ‘apples in a pot and pears on a plate’.

The second level involves the robot. This level is also made up of two sub-levels: the desire sub-level and the robot-goal sub-level. The first sub-level involves the robot’s internal representations of purposes, called *desires*. A key feature of desires is that they are domain-independent, that is, the robot can realise them in different domains (third level discussed below). This allows the robot to satisfy the users’ purposes in a flexible way under different conditions. Desires are a key element of the framework as they drive the downstream learning processes of the robot, for example to acquire world representations, goals, skills, and forward models.

A desire can be hardwired by a designer into the robot, in which case it is called a *need*, so as to best align it with a corresponding purpose. These would be related to the phylogenetic motivations in natural organisms. An example of need is ‘the battery is sufficiently charged’: this is an example of *homeostatic need*. Another example is ‘having a set of images on fruits’ to train an internal classifier: this is an example of *epistemic need* (i.e. a need related to an intrinsic motivation). Notice that in general a need can be seen as a purpose of the designer that was hardwired in the robot. Alternatively, a desire can be autonomously acquired by the robot through suitable learning processes, in which case it is called a *mission*, with the aim to align it as much as possible with the purpose of a user. An example of mission is for example ‘fruits sorted into different containers’. Notice how the learning of the mission involves some guidance from a hardwired criterion deriving from a designer’s purpose. This criterion might be a need or it might be implicitly encoded in the learning mechanism of the robot. This need or mechanism might for example drive the robot to suitably interact with the user in order to encode his/her purpose as a mission.

The second robot sub-level involves different *robot-goals*. These robot-goals are specific observation representations that correspond to the desire in different domains. The robot-goals are thus robot’s internal representations of desired domain states (third level).

The third level is the *domain level* that involves different possible domains (here a domain is intended as the physical/social *environment* external to the robot, plus the *sensorimotor body* of the robot).

Each robot-goal might correspond to a different state in each different domain the robot might encounter, and which are called *state-goals*. Also a user-goal might have a correspondence with a different state-goal in different domains (we will not expand the fact that, similarly to the robot, also the designer/user has two internal sub-levels, one of which encodes user-goals). Importantly, if a robot’s desire is *aligned* with the corresponding designer/user’s purpose, then they correspond to the same state-goal within each domain. This is called here *triangular alignment*.

We shall see that the framework allows the description of several challenges for robotics. After the formalisation of the

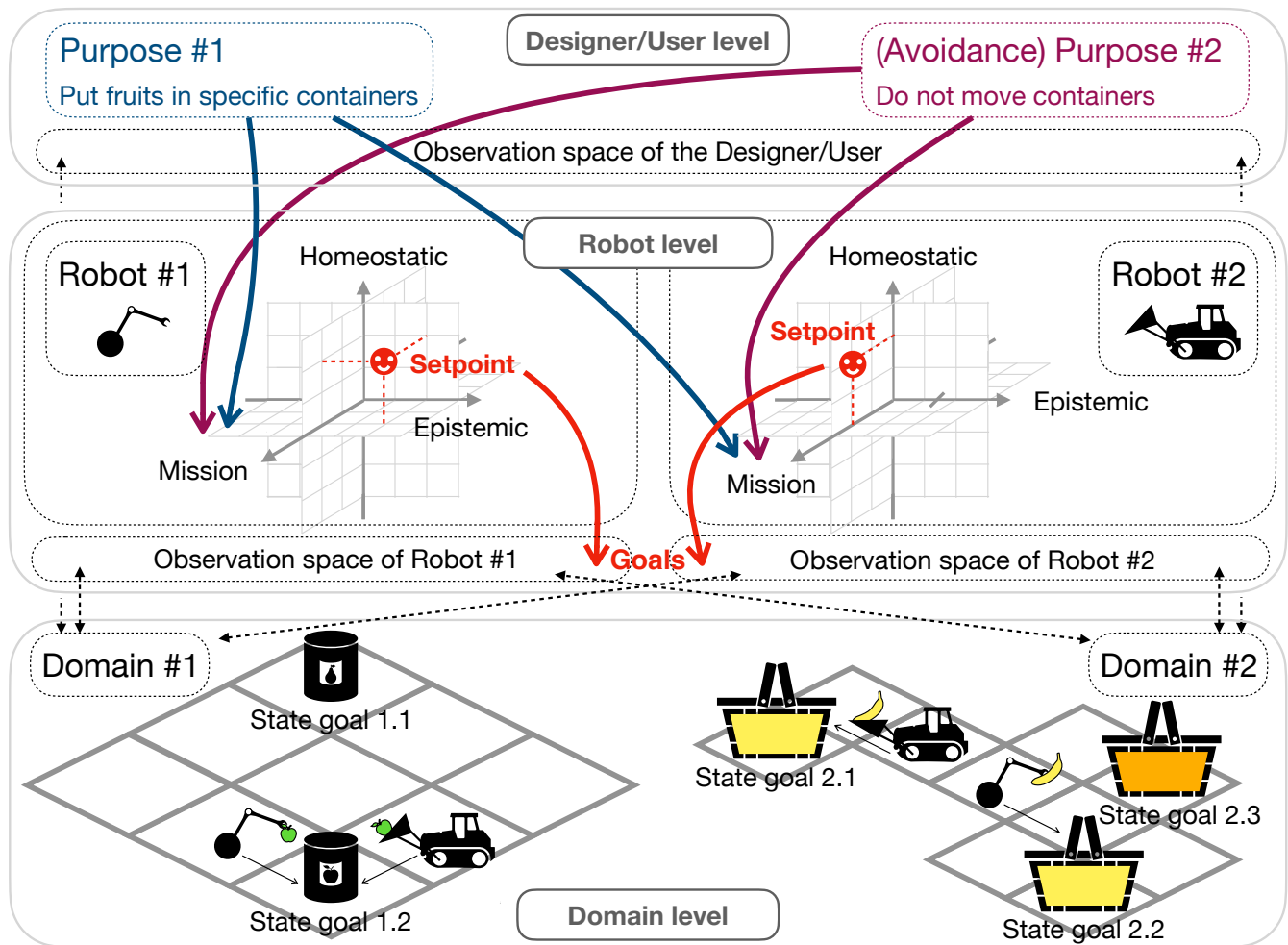


Fig. 1: **General scheme of the purpose framework.** (Top) At the designer/user level, robot-independent, domain-independent, purposes are defined, which constitute the desiderata from the robots. (Middle) At the robot level, each robot is equipped with a motivational space encompassing different desires, for example a mission associated to the user’s purpose (e.g.: ‘sort fruit in different containers’), an epistemic need to acquire information on fruits and containers, and a homeostatic need to keep the battery charged. For simplicity here each desire is represented by one space dimension but they commonly involve many. The dimensions of one desire form a distinct desire-spaces, but then are combined to form a whole motivational space. Each robot has a desirable ‘setpoint’ within its motivational space (blob marked by the smiley face). For instance, robot 1 is curious while robot 2 is not. Robots are also equipped with an observation space which reflects their perception of the state of domains. (Bottom) Reaching different state-goals in different domains satisfy the same desires when they correspond to the same set of points within the robot’s motivational space. For instance, here, filling the pear container with pears (Goal 1.1), the apple container with apples (Goal 1.2), the upper-left yellow container with bananas (Goal 2.1), the bottom-right yellow container with bananas (Goal 2.2), and the orange container with oranges (Goal 2.3), all satisfy the same mission related to Purpose 1: ‘A container filled in with one type of fruit’.

purpose framework, we start to expand on the following ones:

- *Purpose-desire alignment problem*: how to ensure that the robot is endowed with needs, or autonomously learns missions, that are aligned with the related purposes?;
- *Purpose-goal grounding problem*: given a desire, how can the robot acquire goals in different domains to better satisfy it?
- *Purpose-based attention and exploration*: how could purpose bias active perception and exploration to maximise action effectiveness and learning speed?
- *Arbitration of desires*: how can the robot decide which

desire to attend among multiple ones?

- *Multi-robot problem*: how can multiple robots contribute collectively to address a purpose?

IV. FORMALISATION OF THE PURPOSE FRAMEWORK

This section presents the formalised framework on purpose and related concepts. Figure 2 summarises the main elements of the formalisation. The formalism is presented from the point of view of an external observer, for example the researcher, who observes different designers/users of the robots, the robot’s controller, and the world formed by different domains

(here domains include the sensorimotor part of the robot body and the environment external to the robot).

We formalise the purpose framework focusing on a goal-based perspective, but also building on the formalism of Markov Decision Processes (MDPs) used in reinforcement learning (RL) [15].

a) *Symbols*: In the notation that follows, small letters indicate one element of a set while capital letters indicate sets. Sub-script indexes are used for enumerating elements (e.g., O_c and O_3 could be used to indicate the observations O of robot $c = 3$). Over-script indexes are used as part of the symbol to use the same symbol to indicate a subset (e.g., O^G could indicate the subset of observations forming a goal). Functions are indicated with the letter f and are specified with an over-script related to the involved sets (e.g., $f^{O-\Omega}$ establish a correspondence between robot observations O and desire points Ω). Given a discrete set S , $\Delta(S)$ indicates the *probability simplex* over S , that is, the set of all possible probability distributions over S . We sometimes refer to sets as ‘spaces’ to hint to the fact that they have a certain structure (e.g., an observation set O can be considered a space where different observations have similarities).

b) *Main elements of the formalism*: The main elements of the three levels of the formalisation are denoted as follows. Different *users/designers* are denoted by means of index $h \in H$ (symbol memo: ‘h’ stands for human; henceforth we refer to ‘user’ to refer to both designers/users when not necessary to distinguish them). Different collaborative robots (‘*cobots*’) are denoted with the index $c \in C$. Different *domains* are denoted with the index $d \in D$.

c) *Domains*: Time is represented as discrete steps $t \in \{0, 1, 2, \dots\}$. Each domain d is characterised by states $s_d \in S_d$, where S_d forms the domain *state space*. The domain is also characterised by a transition function $f_{d,c}^{S^A-S} : S_d \times A_c \mapsto \Delta(S_d)$. This function returns the probability that a state $s_{d,t+1}$ is produced at time $t+1$ if at time t the domain is in state $s_{d,t}$ and the robot c performs the action $a_{c,t} \in A_c$ chosen from its *action set* A_c .

d) *User observations, purposes, and goals*: The user h has *observations* $q_h \in Q_h$ (symbol memo: ‘Q’ is a letter similar to the ‘O’ denoting the robot’s observations considered below). Q_h is the user’s *observation space*, formed by sensory vectors produced by the observation function $f_h^{S-Q} : S_d \mapsto \Delta(Q_h)$ (note that ‘observations’ could be representations more abstract than the perceptual representations considered here for simplicity).

The user has also different *purpose spaces*. A *purpose space* $P_{h,i}$, indexed with i , is a set formed by points $p_{h,i} \in P_{h,i}$. It is basically a representation space that allows a delimitation of the areas that are established as purposes. So the purpose space is not a ‘space of purposes’, as the name might suggest, but rather a space where a purpose is represented. For example, if a user has the purpose of having ‘some apples in a basket on the table’, the purpose representation space could have two dimensions: the number of apples in the basket, and the distance of the basket from the table. The observation points are associated to the purpose points by a ‘user observation function’ $f_{h,i}^{Q-P} : Q_h \mapsto P_{h,i}$.

A *purpose* is defined within the purpose representation space on the basis of a utility function. The *utility function* $f_{h,i}^{P-U} : P_{h,i} \mapsto U_{h,i}^P$ that maps the purpose space elements $p_{h,i} \in P_{h,i}$ to their utility $u_{h,i}^P \in U_{h,i}^P \subseteq \mathbb{R}$, for example having a range $u_{h,i}^P \in [0, 1]$. Thus, the desirability of the purpose space elements indicating the number of apples in the basket, and the position of the basket in space, might increase with the increasing number of apples and decrease with the distance of the basket from the desired location. The *purpose* is specifically defined as the subset of points of the purpose space, $P_{h,i}^U \subset P_{h,i}$, that have a utility different than zero, that is: $P_{h,i}^U = \{p_{h,i} \in P_{h,i} \mid f_{h,i}^{P-U}(p_{h,i}) \neq 0\}$ (symbol memo: purposes and desires are marked by the super-script U hinting to the fact that they are formed by points of their space having an intrinsic utility). ‘Avoidance purposes’ could be defined on the basis of a negative utility.

The i^{th} purpose $P_{h,i}^U$ of user h corresponds to different user goals $Q_{h,i,d}^G$ in different domains $d \in D$. In particular, each user-goal in a domain, depending on a certain purpose $P_{h,i}^U$, is formed by a subset of user’s observations: $Q_{h,i,d}^G = \{q_h \in Q_h \mid f_{h,i}^{Q-P}(q_h) \in P_{h,i}^U\}$.

In turn, each user’s goal $Q_{h,i,d}^G$ corresponds to a state-goal, which is a subset of the domain states $S_{i,d}^G$ determined by the user observation function: $S_{i,d}^G = \{s_{i,d} \in S_{i,d} \mid f_h^{S-Q}(s_{i,d}) \in Q_{h,i,d}^G\}$.

As an example, consider this situation. The purpose might be ‘one-to-three fruits in a container’. In two domains, this purpose might correspond to two different goals, for example the user’s visual sight of ‘one-to-three apples in a basket’ and ‘one-to-three pears in a carry-bag’. In turn, these sets of user’s observation goals correspond to specific sets of physical states in the two domains.

The user can have several purpose spaces $P_{h,i} \in P_h$, where P_h is the set of all such spaces, and related purposes $P_{h,i}^U \in P_h^U$, where P_h^U is the set of all purposes. For example, the user might have a first purpose ‘one-to-three fruits in a container’ and a second purpose ‘two-to-five children amused’.

e) *Robots’ observations, desires, and goals*: Analogously to the user, the robot c has *observations* $o_c \in O_c$. O_c is the robot’s *observation space*, formed by sensory vectors produced by the observation function $f_c^{S-O} : S_d \mapsto \Delta(O_c)$.

The robot has different *desire spaces*, in general defined similarly to the user’s purpose spaces (note that, as for purposes, a point of the desire space is not formed by desire points; rather, the it is formed by points of which a subset can constitute a desire). A *desire space* $\Omega_{c,i}$, indexed with i , is a set of points $\omega_{c,i} \in \Omega_{c,i}$ having a certain number of dimensions. The desire space points are associated with the robot’s observation points by the *robot’s observation function*: $f_{c,i}^{O-\Omega} : O_c \mapsto \Omega_{c,i}$.

Among desires, hardwired needs and learned missions are defined differently, so we can consider for them respectively mission spaces M and need spaces N in place of the general desire spaces Ω . Needs are defined analogously to the purposes of users. In particular, the need space is characterised by a *utility function* $f_{c,i}^{N-U} : N_{c,i} \mapsto U_{c,i}^N$ that maps the need space elements $n_{c,i} \in N_{c,i}$ to their utility level $u_{c,i}^N \in U_{c,i}^N \subseteq \mathbb{R}$,

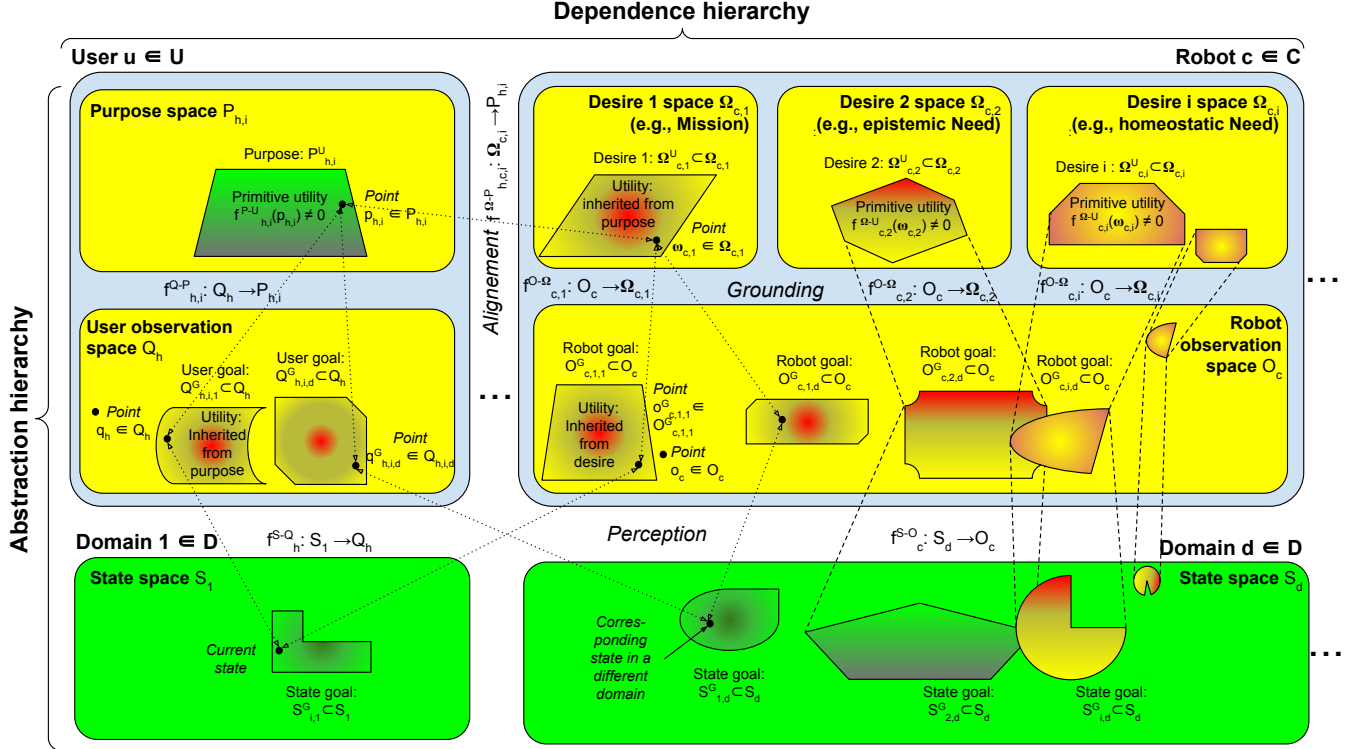


Fig. 2: **The main elements of the purpose framework, including the main elements of the formalism.** The framework involves different domains, but here only two are represented (in green). The user encompasses several purpose spaces (here only one is represented, in yellow), each one abstracting over the user’s observations, and including a purpose. The purpose colour gradient indicates a different-from-zero utility over its elements. Each purpose corresponds to a different ‘user-goal’ (set of observations) for each different domain. The user-goals correspond to sets of states (‘state goals’) in the different domains. There can be multiple robots serving the user’s purposes, but here only one is represented. The robot encompasses several desire spaces, but here only three are represented. Here one desire, the mission, corresponds to the purpose, while the others are hardwired needs. Each desire space abstracts over the robot domain observations. Each desire corresponds to a different ‘robot-goal’ (set of observations) for each different domain. The robot-goals correspond to sets of states (‘state-goals’) within the different domains. The dotted lines in the figure illustrate the constraint that, for the robot to be useful for the users, some/most/all points of each robot’s desire should correspond to the points of the users’ related purpose, via correspondences to the same points of the state-goals in the different domains.

for example having a range $U_{c,i}^N \in [0, 1]$. A *need* is hence defined as the subset of points of the need space, $N_{c,i}^U \subset N_{c,i}$, that have a ‘primitive’ utility different than zero for the robot: $N_{c,i}^U = \{n_{c,i} \in N_{c,i} \mid f_{c,i}^{N-U}(n_{c,i}) \neq 0\}$. ‘Avoidance needs’, for example relevant for ethical purposes, could be defined on the basis of a negative utility.

Missions are instead defined as dependent on purposes. In particular, the points of a mission space, $m_{c,i} \in M_{c,i}$, correspond to the points of the user’s purpose space, $p_{h,i} \in P_{h,i}$, through an *alignment function*: $f_{h,c,i}^{M-P}: M_{c,i} \mapsto P_{h,i}$. The mission is hence formed by the points of the mission space that correspond to a point of the purpose in the purpose space: $M_{c,i}^U = \{m_{c,i} \in M_{c,i} \mid f_{h,c,i}^{M-P}(m_{c,i}) \in P_{h,i}^U\}$. Importantly, each point $m_{c,i}^U$ of the mission inherits the utility of the corresponding point of the user’s purpose: $f_{c,i}^{M-U}(m_{c,i}^U) = f_{h,i}^{P-U}(f_{h,c,i}^{M-P}(m_{c,i}^U))$.

The i^{th} desire $\Omega_{c,i}^U$ of robot c corresponds to different robot-

goals in different domains $d \in D$. In particular, each robot-goal $O_{c,i,d}^G$ is formed by a subset of robot observations as follows: $O_{c,i,d}^G = \{o_c \in O_c \mid f_{c,i}^{O-\Omega}(o_c) \in \Omega_{c,i}^U\}$. Each point $o_{c,i}^U$ forming robot-goal i inherits the utility of the desire point to which it corresponds: $f_{c,i}^{O-U}(o_{c,i}^U) = f_{c,i}^{\Omega-U}(f_{c,i}^{O-\Omega}(o_{c,i}^U))$

In turn, each robot’s goal $O_{c,i,d}^G$ corresponds to a state goal, which is a subset of the related domain states $S_{i,d}^G$ determined by the robot’s observation function: $S_{i,d}^G = \{s_{i,d} \in S_{i,d} \mid f_c^{S-O}(s_{i,d}) \in O_{c,i,d}^G\}$ (we shall see below that $S_{i,d}^G$ should be the same for the user and the robot).

f) Priority of desires and motivation space: Different desires might have different priorities, meaning that some desires might be considered more important than others. One possibility is that desires are organised into a ‘hard hierarchy’ where some desires have to be fully satisfied before other ones influence the robot behaviour. An example of this is that of safety and ethical desires vs. other desires. For example, a

robot might have the desire to not hurt people or damage objects, and this desire should be fulfilled at a maximum level while a mission to clean the house is accomplished.

Another possibility of prioritisation between desires is based on the concept of *motivational space* (Figure 1-Middle). A motivational space is defined here as the Cartesian product of a sub-set or all the desire spaces, e.g.: $\Lambda_c = \Omega_{c,1} \times \Omega_{c,2} \times \Omega_{c,i} \times \Omega_{c,|\Omega|}$, where $|\Omega|$ is the total number of desires. The motivation space has a number of dimensions equal to the sum of the number of dimensions of the need spaces involved. The motivation space can have a utility function $f_c^{\Lambda-U} : \Lambda_c \mapsto U_c^\Lambda$. This function maps the motivation space elements $\lambda_c \in \Lambda_c$ to their desirability level $u_c^\lambda \in U_c^\Lambda \subseteq \mathbb{R}$, for example having a range $u_c^\lambda \in [0, 1]$. This utility function generates a set of points Λ_c^U , within the motivational space, that have a utility different from zero and thus form a *motivational setpoint* within it. Such a utility function might for example allow the robot to mediate between different desires in a soft way, as further discussed in a section below.

V. COMPLEMENTS TO THE THEORY

A. User's and robot's state goals

Importantly, a certain purpose $P_{h,i}^U$ and the related robot desire $\Omega_{c,i}^U$ correspond to different observations (goals) of the user and the robot, respectively $Q_{h,i,d}^G$ and $O_{c,i,d}^G$. This is due to the fact that the user and the robot perceive the world with different sensors. However, the percepts of the two agents should tend to be grounded on the same world states as the robot should aim to achieve results in the environment that accomplish the user's purpose. In particular, there should be a one-to-one correspondence between the two sets $P_{h,i}^U$ and $\Omega_{c,i}^U$ established by the mediation of the states $s_{i,d}^G \in S_{i,d}^G$ in the domain d . Alternatively, their correspondence might be only partial but still allow the robot to serve to a good extent the user's purpose. Metrics could be developed to measure such an overlap, possibly also employing the utility functions to weight the areas of the correspondence or lack thereof.

B. Dynamic priorities and utility functions

During its operation, the robot might be employed to accomplish different desires. For example, at different times of the day, or in different context, the robot should accomplish different missions related to different users; or should assign a different importance to some missions (e.g. a task relevant for a user) and other needs (e.g., enhancing the importance of safety needs in a context with children or animals). The best way to deal with this dynamics of desires is to suitably manipulate their priorities. Indeed, this supports the possibility to change the *relative importance of desires*. Instead, it seems not useful to (also) change the utility functions to this purpose (utility functions that for missions are inherited from the purpose). This because the utility functions are suited to *establish the relative importance of points forming a desire*, rather than the relative importance between different desires.

C. Reward and value functions

The robot might employ reward functions $R_{c,d} : O_c \times A_c \times O_c \mapsto \Delta(\mathbb{R})$ that return a probability distribution of the reward in correspondence to the explored states and performed actions. Reward functions present some differences with respect to utility functions [16]. Utility functions are usually employed to assign a value to one observed state. Instead, reward functions are commonly used to assign a value to trajectories of states and actions. As such, they allow to evaluate 'changes' of states and the actions causing them. On the other hand, utility functions are commonly associated to goals, as done here.

If the robot stores goals, these in turn can be used to guide behaviour. In particular, goals allow the robot to have information on the desirable observations (states), an information in addition to utility and reward functions. Such information can be exploited in different ways, for example to furnish a guidance towards the accomplishment of the goal itself. For example, the robot can possibly attempt to decompose a 'compound goal' into partially independent sub-goals [17].

The goals deriving from desires and utility allow the generation of reward functions. As often done within the RL framework (e.g., for options), goals can then be used to generate a 'pseudo reward' function that produces a reward of 1 when a goal is achieved, and 0 otherwise.

The robot might also be endowed with *expected utility functions* similarly to the evaluation functions in the RL framework. The expected utility functions can involve either the domain-specific robot representations (e.g., the observation spaces) or the domain-independent desire spaces. In the former case, the expected utility function is defined or learned in relation to the utility of robot-goal points similarly to what is done in RL, for example based on the future expected utility considered as a reward in RL.

Alternatively, the expected utility function could be defined over the domain-independent desire space. In this case one should define a metrics to weight the distance from the desire points forming the desire. This metric would change depending on the specific domain considered.

The expected utility could be learned by the robot, for example based on RL techniques [15]. In addition, the initial values of some states of the evaluation function might be hardwired to facilitate the discovery of the policy to achieve points of the desire or corresponding goals in specific domains.

D. Different types of desires

1) *Primitive and learned desires*: we have seen that there are two main classes of desires that the robot could have. The two classes, which depend on the origin of the desires themselves from the user or from the robot, involve *primitive (innate) needs* and *learned missions*. These two classes are now considered in more detail.

a) *Primitive desires, needs*: Needs are primitive desires that are defined by the designer. These needs are very important to implement relevant functions. A first type of functions has to do with the general behavioural operation of the robot. For example, the algorithms written in the robot represent

primitive needs (if they cannot be modified by the robot during learning), for example an obstacle avoidance reflex. A second more explicit type of primitive need involves a condition when a purpose is directly translated by the designer into the robot, for example the need to have a ‘high level of battery charge’. A third important type of functions of primitive needs involves the ethical constraints that the designer might want the robot to respect. This needs could have a high ‘priority’ with respect to other needs, see below.

b) Learned desires: missions: Learned missions are desires that the robot learns during its life (operational). There could be different types of missions. A first type missions are directed to encode the user’s purpose. A second type of missions involve desires that the robot self-generates as instrumental to accomplish other needs or missions. This requires notable power and flexibility of the robot’s algorithms because, contrary to the previous type, these desires have to be created from scratch (in particular their spaces).

2) Taxonomies of desires related to motivation classes: Desire types could also be distinguished based on the traditional distinction between extrinsic motivations, social motivations, and intrinsic motivations. Extrinsic motivations might further generate two sub-types of desires. Note that the taxonomy considered here has a heuristic value but has a blurred nature. For instance, it is possible that the classes that form the taxonomy are not mutually exclusive. As an example, a mission might involve an intrinsic need or a social need.

a) Extrinsic desires: A first type of extrinsic desires involve learned desires that directly aim to satisfy the user’s or designer’s purposes. These have been called ‘missions’. A second type of extrinsic desires could reflect purpose, but be represented by needs hardwired into the robot by the designer.

b) Homeostatic needs: Homeostatic needs are a different type of extrinsic needs. They are analogous to the homeostatic needs of biological agents, mainly involving survival (preservation of physical integrity of the robot; safety for the user) and physical functioning (acquisition of energy and other resources). An example of homeostatic need might involve a need space having three dimensions: the battery level; the integrity level of the robot’s wheels, as measured by a suitable sensor; the integrity level of the robot’s gripper, as measured with a further sensor.

c) Social needs: Another type of needs are social needs. These require the robot to have some engagements with humans, or with other robots, in order to guarantee the achievement of a certain social state. An example of this might be a robot with the need to engage some visitors of a museum.

d) Intrinsic needs: These needs support the robot’s acquisition of new knowledge and skills, for example under the drive of some intrinsic motions. An example of epistemic need space might for example involve two dimensions: the level of competency that the robot has for the skills that it has found to accomplish a certain mission; the level of novelty of the current perceived state [8]. Under Bayesian formulations, uncertainty reduction for any type of internal representation or world model can be an intrinsic need [18], [19].

VI. ILLUSTRATIVE SCENARIO

This section provides the reader with a concrete illustration of the purpose framework. In particular, it presents a scenario where the user of an open-ended learning robot observes it interacting with the world and progressively refines the purposes assigned to it. Figures 3 to 6 sequentially illustrate different steps of this illustrative scenario.

The scenario starts with the user of a newly-acquired OEL robot first turning it on and testing it before assigning it with any purpose. For simplicity, and for the sake of visualisation, we imagine that the designer had programmed the robot to have a *motivational space* encompassing only three dimensions related to three different desires, one dimension per desire: a social need, an energy need, and a mission (initially ‘empty’, that is with a utility for all points equal to zero, and ready to be filled by the user) (Figure 3).

Importantly, it would be useful to also show in the figure the *utility* associated to each point of the axes of the motivational space (a point corresponding to a possible abstract state). However, this would require an additional dimension for each axis that cannot be graphically represented in the figure. In addition, the utility of each desire point should be multiplied by the desire priority, and the ‘prioritised utilities’ would generate an additional dimension indicating the overall desirability of each point of the motivational space: again, this cannot be represented in the figure. To overcome this graphical limitations, we assume that the points on each axis have an increasing utility (possibly weighted by the desire priority) the closer they are to the origin of the axes. The origin point is assumed to have the maximum utility for all desires, but in real scenarios multiple motivation points might have the same prioritised utility.

The *social need space* is very simple. Its 1 dimension is assumed to be formed by only 2 points. One point, which has a maximum utility of 1 for the robot and is at the origin of the axes, corresponds to the states of the environment (domains) in which the robot is in the same spatial position in which it perceives a smiling human. This single point constitutes the robot’s social need. This holds for any domain which, for simplicity, are assumed to be grid worlds featuring the Markov property. The second point in the dimension corresponds to all other possible states in the domains.

The purpose framework allows the definition of an arbitrary number of social needs, and hence social motivation dimensions. For instance, one could add a social dimension for which the robot experiences a high utility when receiving a positive verbal feedback from a human, and another social dimension for which the robot perceives a high utility when experiences joint attention with a human during a collaborative task [20].

The *energy need space* is formed by a continuous dimension ranging in $[0,1]$ and having a utility itself ranging in $[0, 1]$. The points in the domains in which the robot is in a spatial position in contact with a battery charger correspond to the points of the need-space having a positive utility. In particular, the utility depends on the battery level of the robot: the lower the battery level, the higher the utility of the need-point. For example, we can assume that the battery level ranges in $[0-1]$

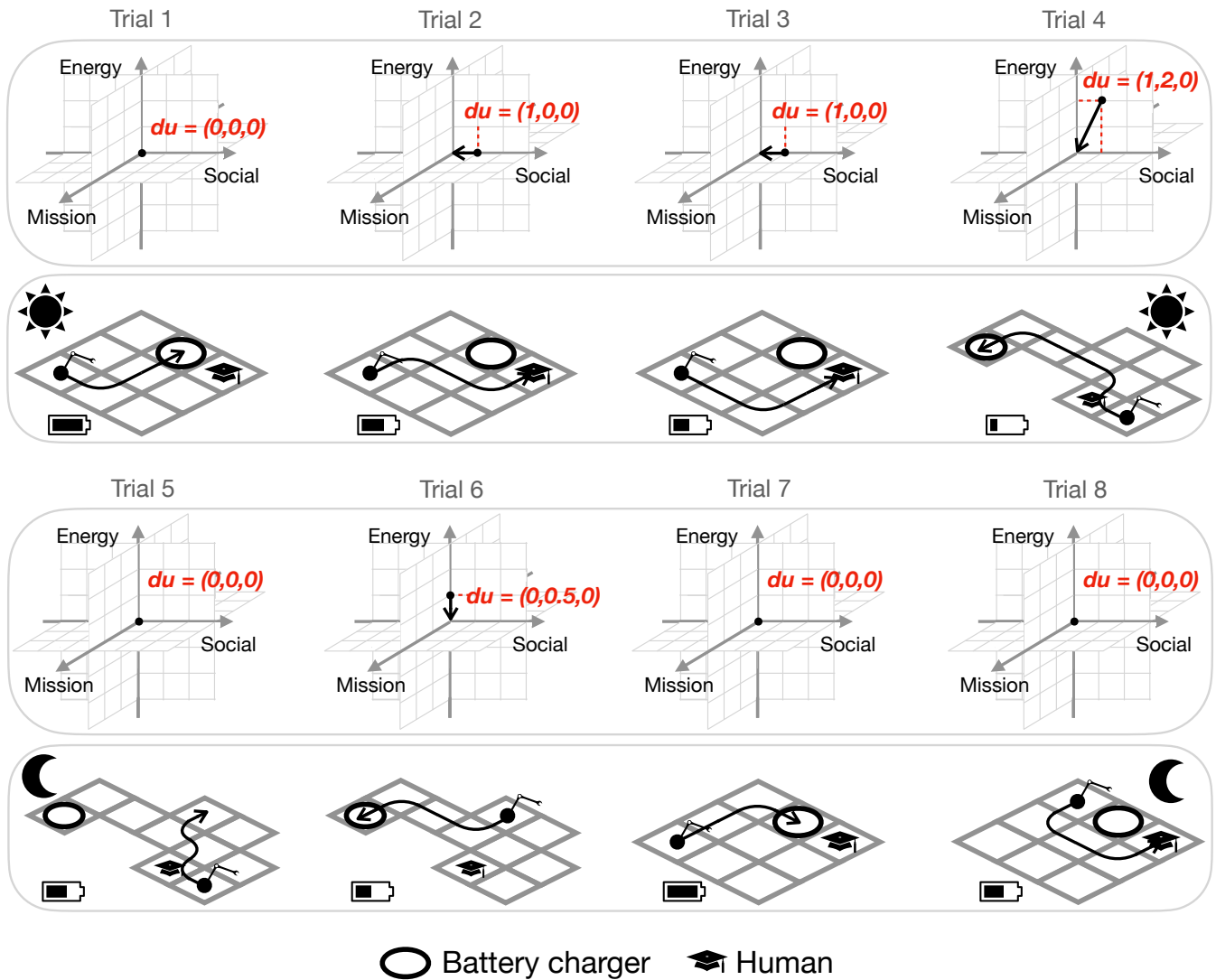


Fig. 3: **Illustrative scenario.** A robot with 3 desires (social need, energy need, mission) performs a series of ‘trials’ in two different domains during day time and night time. Both domains include a human agent and a battery charger. The robot starts with an ‘empty mission’ and thus gets to high-utility states when visiting a human during day time, or when recharging while the battery level is low. The center of each plotted motivational space represents the ‘set-point’, *i.e.*, the ideal point, associated with maximal utility. The vectors represent utility variations resulting from the robots’ actions.

and this corresponds to [0-1] values of the axis, but this only when the robot is in touch with the battery charger; the states in which the robot is not in contact with the charger always correspond to the point equal to 1 of the axis having 0 utility. Thus, for example, if the robot has a battery charged with 0.3, then (assuming a linear utility) the need-space point is 0.3 and has a utility of 0.7 if the robot is in touch with the charger.

The *mission* corresponds to the dimension of the robot’s motivational space where a positive utility is perceived when achieving a point of the corresponding user’s purpose. For example, assume the robot has a sound receiver that can be activated by the robot user, for example by saying “Robot: well done!”, to tell the robot that it has successfully accomplished a purpose point. As for the other dimensions, the robot can progressively discover which states of each domain yield a

positive utility on the mission space points as it corresponds to a purpose point.

The robot forms domain specific goals as follows. Each state within each specific domain, that corresponds to a desire point (of the three desires) having a utility above zero is stored by the robot as a domain-specific goal encoded in the robot observation space and possibly pursued in the future. Importantly, the goal point inherits the utility value of the related desire point.

The learned goals can be used in different ways to acquire the capacity to accomplish them when desired. The robot can use a reinforcement learning process to learn the skills to achieve a given goal in a given domain. This process could be driven, for example, by producing a (pseudo-)reward value ranging in [0, 1] depending on the utility of the goal point experienced [21], [22]. For example, the reward can be 1 when

the robot passes from one domain state to another state that corresponds to an observation forming the goal and having a higher utility. Alternatively, the robot could use the goal to guide a planning process [23]. We now consider more in detailed how these processes work based on specific examples.

First phase of the scenario (Figure 3). The robot performs 8 ‘trials’ (i.e., 8 sequences of actions, or episodes), 4 during day time followed by 4 during night time. Trial 1: the robot first explores Domain 1 and reaches the battery charger. Since its battery is already full, it does not perceive any utility. Trial 2: the robot continues to explore Domain 1 and reaches the human, who smiles at the robot. The robot thus experiences its first social utility and forms a social goal representation. Trial 3: The robot explores a different path in Domain 1, and again reaches the human, who smiles at it, thus social utility is achieved for a second time. Trial 4: the robot now explores Domain 2, passes by another human who smiles at it (social utility of 1, triggering the formation of second social goal), and then reaches a battery charger. Since its battery is nearly empty, it perceives a large energy utility of approximately 1. Trial 5: It is now night time. The robot continues to explore Domain 2. It passes by the human, who does not smile (say she is sleeping). So no social utility. Trial 6: The robot explores yet another path within Domain 2 and reaches the battery charger. Because the battery is half empty, it gets an energy utility of 0.5. Trial 7: The robot gets back to Domain 1 and reaches again the battery charger. No utility since its battery is already full. Trial 8: The robot reaches the human in Domain 1, who is also asleep. No smile and hence no social utility.

Second phase of the scenario (Figure 4). The robot now performs some sort of experience replay, based on the 8 experienced trials. Any kind of off-line learning process could take place at this stage. Nevertheless, for the sake of illustration, we imagine that here the robot is performing some sort of *representation redescription process* [24] so as to make sense of its past observations. Here, we simply imagine that the robot is equipped with model-based processes, and trains a neural network to estimate the reward function of the environment [25]. Moreover, the robot attempts to cluster the effects of its past actions into different contexts, in terms of reward signals observed in different dimensions. This leads to the identification of the following contexts: In the context where its battery level is low (Figure 4-Top), visiting the battery charger yields an energy utility and zero social- and mission-utility. In the context where its battery level is high (Figure 4-Top), visiting the battery charger yields approximately no energy utility, and zero social- and mission- utility. In the context of day time (Figure 4-Bottom), visiting the human yields a social utility, and zero energy - and mission- utility; In the context of night time (Figure 4-Bottom), visiting the human yields zero social, energy, and mission utility.

Third step of the scenario (Figure 5). The user now adds a purpose to the robot. Its mission consists in visiting humans during day time. This translates into a positive utility of 1 on the mission dimension each time the purpose is fulfilled, and zero mission utility the rest of the time. Importantly, while the present paper is agnostic about the specific methods that could be used to enable the robot to encode the purpose into its

mission (e.g., by linguistic feedback from the user), the idea is nevertheless that any method permitting robots to learn to categorise their actions’ effects in different contexts should facilitate the interpretation of purposes.

The robot now performs a new series of 4 trials, two during day time in Domain 1, one during night time in Domain 1, one during night time in Domain 2 (Figure 5). The important thing to note here is that any time a desire point is reached, a positive utility signal is yielded. For instance, during Trial 9, the robot visits the human during day time, thus making it smile, which simultaneously produces a social goal achievement and hence a utility increase of 1, and simultaneously a mission utility increase of 1. Thus, mission utility increases act as a bonus that bias the robot’s behaviour towards desired states of each specific domain, corresponding to the purpose. As previously mentioned, this permits open-ended learning robots to store states that fulfil specific purposes, so as to later treat them as goals for action planning. Interestingly, during Trials 11 and 12, the robot visits the human during night time, which yields neither social nor missions utility variations. This is because the current purpose does not penalise the robot for doing so. It only encourages the robot to visit the human during day time.

Last step of the scenario (Figure 6). After observing the robot during Trials 9-12, the user wants to further bias the robot towards fulfilling its purpose: visiting human during day time. It thus decides to encourage the robot to recharge battery during night time, hoping that this would lead its battery to be most of the time full during the day, and thus reducing the robot’s desires to recharge during day time. The user thus adds a second purpose: ‘Recharge during night time’. Note that the user could later realise that this is a too weak incentive, and could thus decide to further add an avoidance purpose: ‘Do not recharge during day time’.

In addition, the user wants to encourage the robot not to visit the human during night time. It thus adds a third (avoidance) purpose: ‘Do not visit human during night time (Figure6). The user decides to assign a strong negative utility of -10 when this third purpose gets violated. Now the robot is ready to perform a last series of four trials. We contentedly choose Trials 13-16 to represent exactly the same behaviour as Trials 9-12, so that the reader can see what the addition of the two purposes changes to the utility signals in these cases. Trial 13 shows the same utility signals as Trial 9 because the two new purposes are not affected. Trial 14 shows the same utility signal as Trial 10 to illustrate that the second purpose is not sufficient to penalise the robot when it recharges during day time. Trial 15 shows a situation where the robot visits the human during night time, and thus receives a penalty of -10 on the mission dimension. Trial 16 shows a case where the robot both visits the human and recharges during night time. It thus gets -10 plus +1 on the mission dimensions, thus -9 in total.

VII. THE CHALLENGES OPENED BY THE PURPOSE FRAMEWORK

The purpose framework poses a number of challenges. Some of the main ones are now expanded.

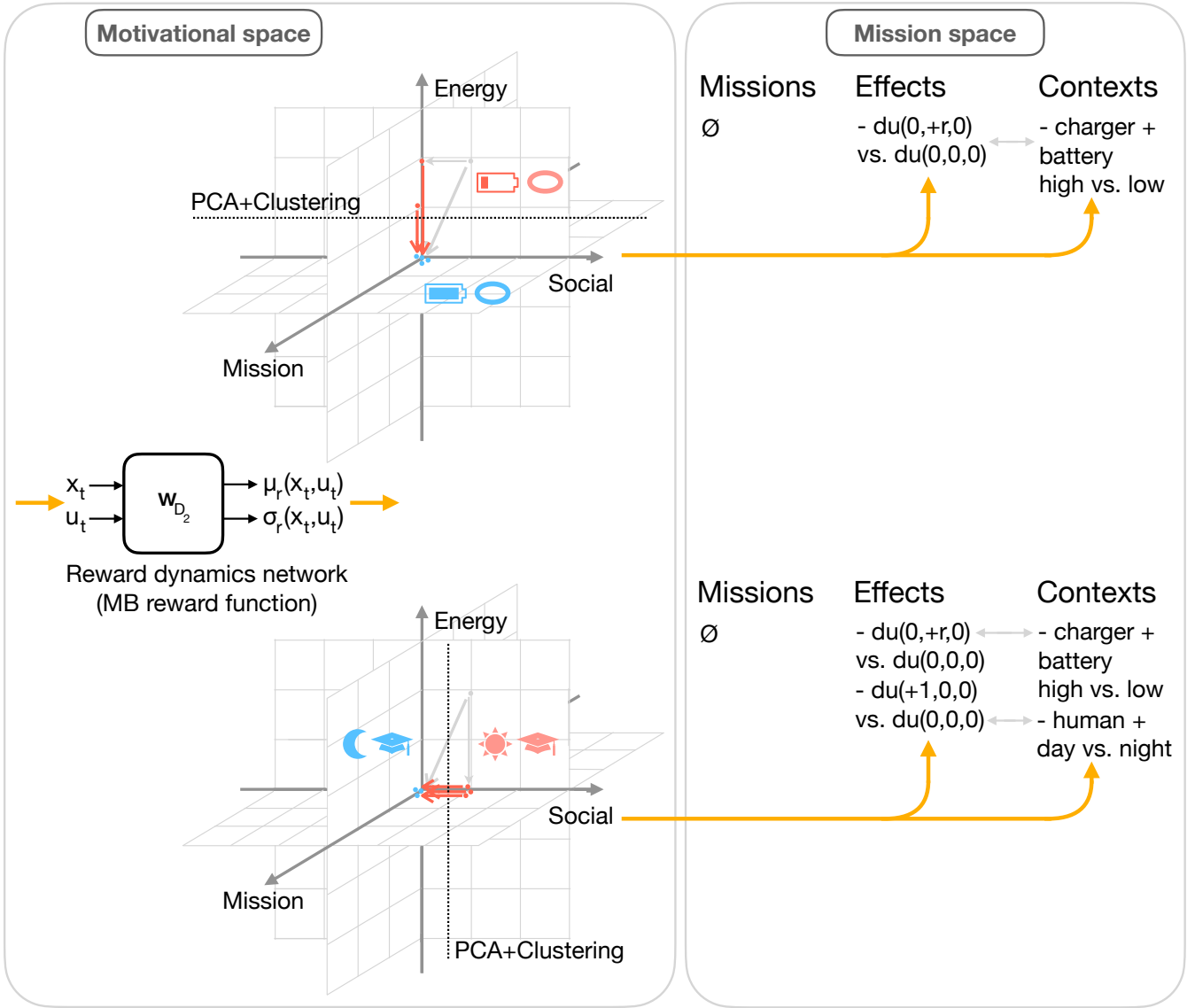


Fig. 4: **Follow-up of the illustrative scenario.** Following the 8 trials performed by the purposeless robot in Figure 3, it performs offline representation re-description by searching for regularities in the experienced observations (e.g.: state, action, state, du (utility variations) quadruplets) so as to learn a model capturing the regularities of the world. The utility variations considered by the model only involve the social and energy needs. For simplicity, here we imagine that the algorithm first performs some sort of Principal Component Analysis (PCA) to find relevant dimensions, and then clustering to find that: (a) the effect of visiting the human (e.g., perceiving a utility-charge smile or not) depends on the context (day vs. night); and (b) the effect of recharging the battery depends on the context (battery high vs. low).

A. The purpose-mission alignment problem

a) *The problem:* This challenge involves ensuring that the robot has a need (mission $M_{c,i}^U$) that corresponds to the user's purpose ($P_{h,i}^U$). This problem involves various elements that can be either hardwired or learned by the robot: (a) *The desire space* $\Omega_{c,i}$, that abstracts the sensory observations in ways that are suitable to establish a good correspondence with the purpose; (b) the need utility function $f_{c,i}^{\Omega-U}(\omega c, i)$, that should possibly correspond to the user's utility function over the purpose space, $f_{h,i}^{P-U}(ph, i)$.

b) *Solution strategies:* There are various possible solutions to this problem. The purpose could be directly hardwired

into the robot in the form of 'need'. As an alternative, a mission should be inferred by the robot by showing it some instances of the state goal corresponding to it in some domains. On this basis, the robot should infer the mission (or part of it depending on the considered domains). A third possibility is that the user and the robot have a purpose space and a mission space that coincide. In this case also the purpose and the mission coincide. This is for example the case (maybe the only existing instance) when the purpose and mission spaces are represented by natural language, in which case the purpose and the mission can be expressed with the same sentences. However, as language is ambiguous and partially subjective,

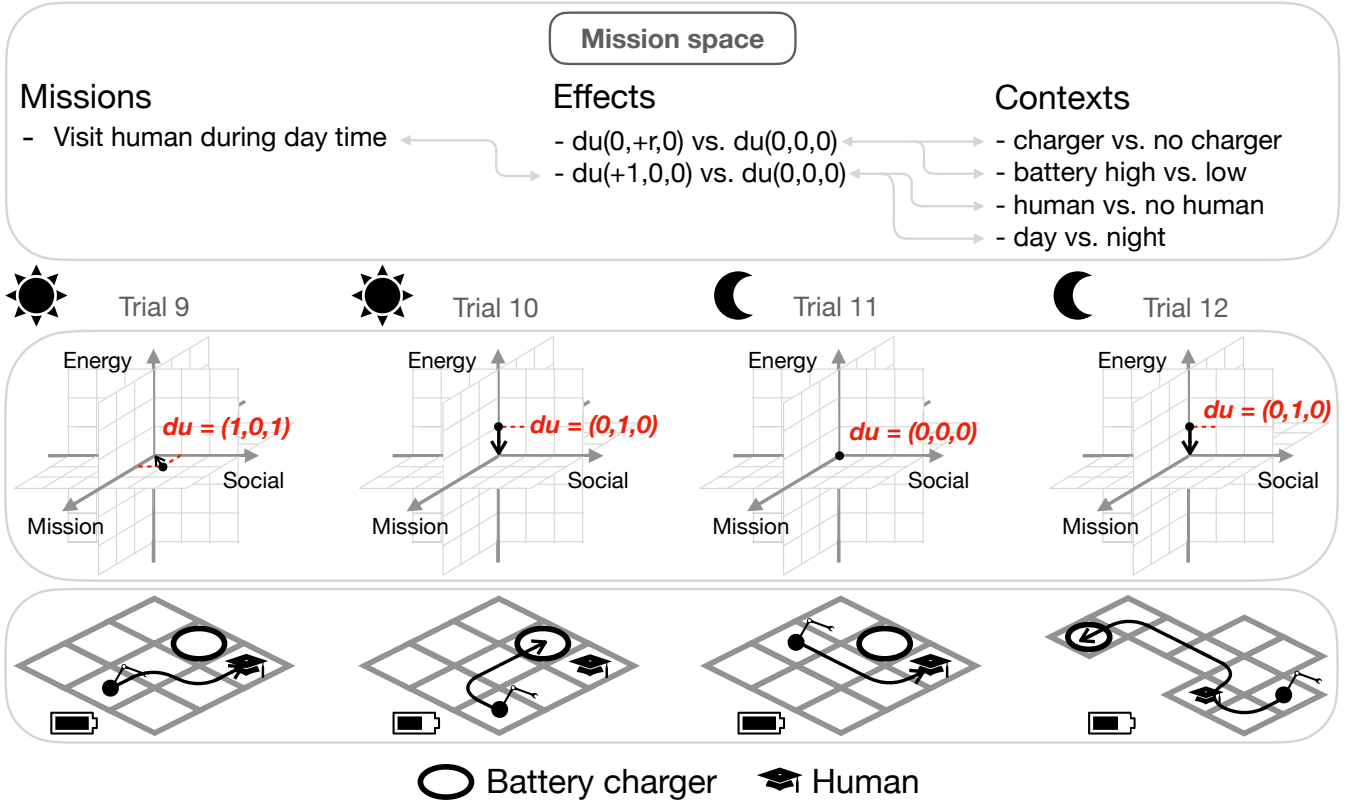


Fig. 5: **Second follow-up of the illustrative scenario.** Now a first purpose is assigned by the user to the robot, which translates into the following mission: ‘Visit human during day time’. The robot interprets this mission thanks to the effects it has learned. As a consequence, during the next four trials (Trials 9-12), the robot will continue to obtain social and energy utility signals, plus a mission utility signal when the purpose is fulfilled.

the user and the robot might still have a different groundings of the purpose/mission.

c) *Two important types of alignment challenges:* There are two important types of alignment challenges. The first class, generating a *RL-like alignment problem*, involves cases where the user is satisfied if the robot discovers *at least one state-goal* $s_{c,i,d}^G$ that fulfils the purpose, and the robot is able to accomplish it with a competence above a certain threshold. This condition can be formalised as follows (assuming for simplicity that any point of the purpose has the same utility for the user):

$$\exists s_{c,i,d}^G : \left(f_{h,i}^{Q-P}(f_{h,i,d}^{S-Q}(s_{c,i,d}^G)) \in P_{h,i} \right) \wedge (R(f_c^{S-O}(s_{c,i,d}^G)) > th_{c,i,d}) \quad (2)$$

where $R(f_c^{S-O}(s_{c,i,d}^G))$ is a reward function indicating the robot competence on the robot goal.

A possible objective function that captures the RL-like alignment problem for a certain domain is one for which the robot is able to achieve a domain state that represent a robot goal point for which it has the highest performance. Formally:

$$\theta^* = \max_{\theta} (f^1(o_{c,i,d} \in O_{c,i,d}^G) \cdot R(o_{c,i,d})) \quad (3)$$

where θ are the robot’s control parameters to be optimised, $o_{c,i,d}$ is an observation that is assumed to be producible by the robot’s controller in the environment, $f^1(o_{c,i,d} \in O_{c,i,d}^G)$

is the function that returns 1 if $o_{c,i,d} \in O_{c,i,d}^G$ and zero otherwise, and $R(o_{c,i,d}) \in [0,1]$ is a function that returns the robot’s competence level (e.g., the probability that the robot’s controller produces $o_{c,i,d}$ within a ‘trial’ lasting a certain length of time).

The second class, generating an *IMOL-like alignment problems*, involves cases where the user is satisfied if the robot can accomplish, with high competence, *every* point of the state goal that fulfil the purpose, or in general as many as possible. This might be for example relevant if the user wants the robot to learn to accomplish a large number of results of a certain type, but s/he will assign specific goal instances (points) only in a later stage. This condition can be represented as follows:

$$\forall s_{c,i,d}^G : \left(f_{h,i}^{Q-P}(f_{h,i,d}^{S-Q}(s_{c,i,d}^G)) \in P_{h,i} \right) \wedge (R(f_c^{S-O}(s_{c,i,d}^G)) > th_{c,i,d}) \quad (4)$$

A possible objective function that captures IMOL-like purpose problems for a certain domain can be expressed as the ratio between: the integral over all observations that correspond to accomplishing the goal, each weighted by the robot’s competence for it; and the integral over all observations that correspond to accomplishing the goal. Formally:

$$\theta^* = \max_{\theta} \frac{\int_{O_{c,i,d}} f^1(o_{c,i,d} \in O_{c,i,d}^G) \cdot R(o_{c,i,d}) \, do_{c,i,d}}{\int_{O_{c,i,d}} f^1(o_{c,i,d} \in O_{c,i,d}^G) \, do_{c,i,d}} \quad (5)$$

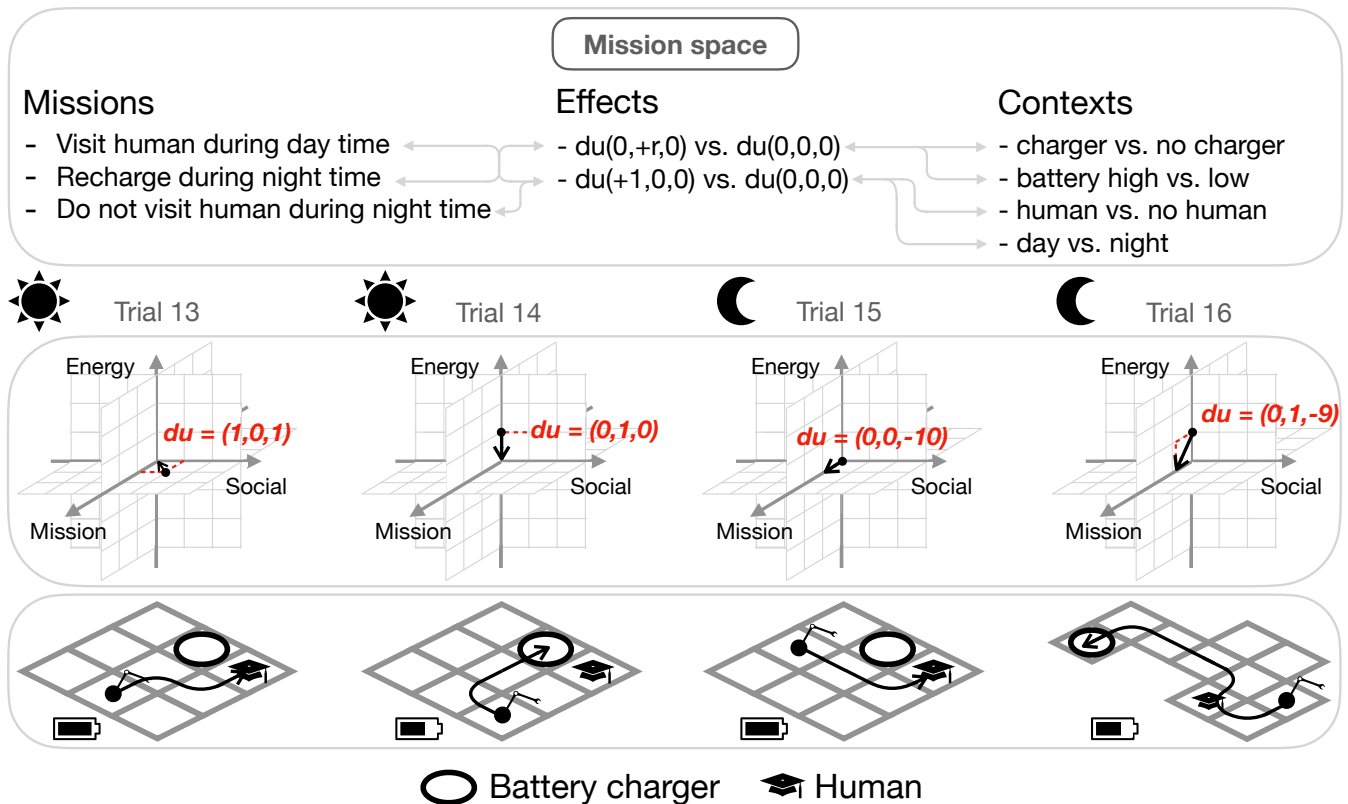


Fig. 6: **Last follow-up of the illustrative scenario.** After Trials 9-12 in Figure 5, the user finds the incentive for the robot to visit the human during day time insufficient. It thus decides to add an avoidance purpose, translated into the additional avoidance mission: ‘Do not visit human during night time’, and associated with a large penalty of -10 in case of violation. In addition, the user encourages the robot to recharge during night time (additional purpose), with a low associated utility ranging in $[0,1]$, and no penalty for recharging during day time. The robot then performs the same series of four trials as in Figure 5, but which now produce different utility signals (Trials 13-16).

where θ are the parameters of the robot controller to be optimised, $f^1(o_{c,i,d} \in O_{c,i,d}^U)$ is the function returning 1 in correspondence to the element $o_{c,i,d}$ belonging to the robot goal $O_{c,i,d}^U$ and 0 otherwise, and $R(o_{c,i,d})$ is a function that returns the robot’s competence level, ranging in $[0,1]$, when it accomplishes the observation $o_{c,i,d}$.

B. The desire-goal grounding problem

a) *Problem:* This problem involves the fact that while the desire is expressed in a domain-independent abstract space, the robot-goals have to be grounded onto specific domain state-goals. For example, a mission might be expressed in written text, for example be ‘fruits sorted into different containers’. To accomplish this mission, the robot should be able to associate it to state-goals that correspond to the specific domains addressed.

b) *Solution:* A possible solution could for example be as follows. The robot could examine the specific domain addressed through computer vision segmentation and identification modules, and detect within it specific objects. The robot might then check if these are the objects that are involved in the mission, say ‘fruits’ and ‘containers’. If this is the case, for example the robot sees two apples, two pears, and two

containers, it might then try to imagine a reconfiguration in space of those objects into a goal that satisfies the mission. This goal might for example correspond to a state of the environment where the two apple are inside the first container, and the two pears inside the second one.

C. How could purpose (missions) influence the active perception and exploration of the world?

a) *Problem:* This problem involves the study of how purpose might influence the robot’s active perception and exploration of the world.

b) *Solution strategies:* An example of solution involves the direction of the robot’s attention system towards ‘objects’ relevant for the mission, detected based on computer vision segmentation and identification modules. For example, if a mission involves objects such as ‘fruits’ and ‘containers’, the pan-tilt camera could be preferentially directed towards these particular objects in the current domain. This attention process might for example facilitate: the acquisition of state-goals to accomplish the mission in different domains; the acquisition of motor skills for manipulating those objects.

D. Arbitration of desires: motivation utility functions

a) *Problem:* What could be the mechanisms used by the robot to balance the importance given to the different desires? An important aspect of this problem is that motivation utility functions can be either static or dynamic during the life of the robot.

b) *Solution:* One example of simple and rigid solution is as follows. The ‘priorities’ given to the different desires are static and hardwired. The different desires are pursued in a strictly hierarchical fashion, meaning that first the robot has to lead the higher-level desires above a certain threshold of utility, and then it can focus on lower level ones.

Alternatively, a more flexible solution could involve the priorities of the different desires dynamically changing depending on the chances of success to accomplish them in the current condition, and the current level of their utility. The robot decides which desire to pay attention to through a decision based on a softmax-based probabilistic selection the current priorities of desires.

E. Multi-robot systems: independent or interdependent

a) *Problem:* One could consider a system composed of multiple robots. The learning of the mission by different robots having different sensors/actuators does not pose a particular problem if the robots act independently. For example, imagining robots having to build different independent parts of an overall structure. In this case the problem does not pose particular difficulties as it can be faced independently by each robot.

A more challenging situation is when a set of robots C have to discover and perform a mission by collaborating. In this case there might be a specialisation of the different robots c , each possibly pursuing a different mission $M_{c,i,d}^U$ that covers a different aspect of the user’s purpose $P_{h,i}^U$.

VIII. CONCLUSIONS

This work gives a theoretical contribution related to open-ended learning (OEL) robots. These are robots able to autonomously acquire skills and knowledge through a direct interaction with the environment, in particular by relying on the guidance of intrinsic motivations and self-generated goals. OEL robots have a notable application relevance as they can use the autonomously acquired knowledge to accomplish tasks relevant for human users. However, an important problem of OEL is that robots explore any possible experience deemed interesting thus acquiring a shallow knowledge on all skills that is of little utility for accomplishing specific classes of user’s tasks.

Here we proposed a possible solution to this problem that pivots on the novel concept of ‘purpose’. Purposes indicate what the designer and/or user want from the robot, for example the accomplishment of specific goals or all possible goals of a certain class. The robot learns an internal representation of the users’ purposes (‘missions’). Missions allow the robot to focus its open-ended exploration towards the acquisition of knowledge relevant to accomplish the purposes. In addition to learned missions, the robot can be also be endowed with

hardwired ‘needs’ by its designer. Needs can ensure that the robot fulfils other important objectives while it pursues its missions, e.g. homeostatic and social needs, for instance keeping its battery charged and avoid damaging humans and itself during actions. Missions and needs are called ‘desires’ and together they form the robot’s ‘motivational space’ that regulates its behaviour and learning.

Thus, we first formalised the concept of purpose by proposing a three level motivational hierarchy that involves: (a) the externally imposed user/designer purposes, corresponding to specific different user-goals in different domains; (b) the domain independent robot internal representations of objectives (‘desires’), some learned based on the purpose and others hardwired (e.g., homeostatic, epistemic, social needs); these correspond to different robot-goals in different domains; (c) specific domain dependent state-goals that should correspond to purposes and desires that are ‘aligned’.

Second, we highlighted key challenges that emerge by employing the purpose framework in robots, and started to discuss how these could be addressed. The ‘purpose-desire alignment problem’ requires to ensure that the needs and missions are aligned with their related purposes. The ‘purpose grounding problem’ requires the robot to acquire goals in different domains to accomplish desires. The ‘purpose-based attention and exploration’ should ensure that the robot performs active perception and exploration maximising the acquisition speed of relevant information. The ‘arbitration of desires’ should dynamically ensure a suitable prioritisation of different desires. The ‘multi-robot problem’ should provide for different robots to suitably coordinate to collectively accomplish the same purpose.

Overall, the approach enables robots to learn, in an autonomous but also focused way, domain-specific goals and skills that meet the desires of the designer/user. Future work should now leverage the framework to develop specific means to address all the challenges highlighted by the framework.

ACKNOWLEDGEMENTS

We thank Olivier Sigaud for feedback on the manuscript.

AUTHORS’ CONTRIBUTION

GB contributed with the theoretical idea on user-robot ‘alignment’, contributed to develop and revise the theoretical framework, wrote the formalisation sections, and revised all sections. RD came up with the original idea of purpose as well as the three level motivational hierarchy, conceived and contributed to develop the whole theoretical framework, contributed to the construction of the formalisation, and revised all sections. EC contributed to develop and revise the whole theoretical framework and the formalisation, and contributed to revise all sections. MK contributed to revise the theoretical framework, conceived and wrote the illustrative scenario (and figure 1), and contributed to revise all sections. AR participated in the development of the theoretical framework. VGS contributed to develop and revise the whole theoretical framework, wrote the first version of the introduction, and contributed to revise all sections.

REFERENCES

- [1] V. G. Santucci, G. Baldassarre, and M. Mirolli, "Which is the best intrinsic motivation signal for learning multiple skills?" *Frontiers in Neurobotics*, vol. 7, no. 22, pp. e1–14, 2013.
- [2] S. Doncieux, D. Filliat, N. Díaz-Rodríguez, T. Hospedales, R. Duro, A. Coninx, D. M. Roijers, B. Girard, N. Perrin, and O. Sigaud, "Open-ended learning: A conceptual framework based on representational redescription." *Frontiers in neurobotics*, vol. 12, no. 59, pp. e1–6, 2018.
- [3] O. Sigaud, G. Baldassarre, C. Colas, S. Doncieux, R. Duro, N. Perrin-Gilbert, and V. G. Santucci, "A definition of open-ended learning problems for goal-conditioned agents," arXiv, Doi: 10.48550/ARXIV.2311.00344.
- [4] E. Cartoni, D. Montella, J. Triesch, and G. Baldassarre, "An open-ended learning architecture to face the real 2020 simulated robot competition," arXiv preprint arXiv:2011.13880v1, 2020.
- [5] K. Seepanomwan, V. G. Santucci, and G. Baldassarre, "Intrinsically motivated discovered outcomes boost user's goals achievement in a humanoid robot," in *Joint IEEE International Conference on Development and Learning and Epigenetic Robotics*, 2017, pp. 178–183, 18-21/09/2017, Lisbon, Portugal.
- [6] C. Brian, *The Alignment Problem: Machine Learning and Human Values*. New York: W.W. Norton & Company.
- [7] M. Khamassi and R. Chatila, "Strong or weak alignment of ai systems with humans values?" *T.B.D. or ArXiv link for the moment*, 2024.
- [8] G. Baldassarre and M. Mirolli, *Intrinsically Motivated Learning in Natural and Artificial Systems*. Berlin: Springer, 2013.
- [9] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual lifelong learning with neural networks: A review," *Neural Networks*, vol. 113, pp. 54–71, 2019.
- [10] M. Ring, "Continual learning in reinforcement learning environments," Ph.D. dissertation, 1994.
- [11] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Proceedings of the 26th annual international conference on machine learning (ICML)*, 2009, pp. 41–48, 14-18/07/2009, Montreal, Quebec, Canada.
- [12] P.-Y. Oudeyer, F. Kaplan, and V. Hafner, "Intrinsic motivation systems for autonomous mental development," *IEEE transactions on evolutionary computation*, vol. 11, no. 6, 2007.
- [13] G. Baldassarre, "What are intrinsic motivations? a biological perspective," in *Proceedings of the International Conference on Development and Learning and Epigenetic Robotics*, 2011, pp. E1–8, Frankfurt am Main, Germany, 24–27/08/2011.
- [14] A. Barto, M. Mirolli, and G. Baldassarre, "Novelty or surprise?" *Frontiers in Psychology – Cognitive Science*, vol. 4, no. 907, pp. e1–15, 2013.
- [15] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 1998.
- [16] E. Hubinger, C. van Merwijk, V. Mikulik, J. Skalse, and S. Garrabrant, "Risks from learned optimization in advanced machine learning systems," arXiv preprint, 2019, available online at: <https://arxiv.org/pdf/1906.01820.pdf>. [Online]. Available: <https://arxiv.org/abs/1906.01820>
- [17] G. Baldassarre, W. Lord, G. Granato, and V. G. Santucci, "An embodied agent learning affordances with intrinsic motivations and solving extrinsic tasks with attention and one-step planning," *Frontiers in Neurobotics*, vol. 13, no. 45, pp. e1–26, 2019.
- [18] K. Friston, F. Rigoli, D. Ognibene, C. Mathys, T. Fitzgerald, and G. Pezulo, "Active inference and epistemic value," *Cognitive neuroscience*, vol. 6, no. 4, pp. 187–214, 2015.
- [19] T. Taniguchi, S. Murata, M. Suzuki, D. Ognibene, P. Lanillos, E. Ugur, L. Jamone, T. Nakamura, A. Ciria, B. Lara *et al.*, "World models and predictive coding for cognitive and developmental robotics: frontiers and challenges," *Advanced Robotics*, pp. 1–27, 2023.
- [20] M. Khamassi, G. Velentzas, T. Tsitsimis, and C. Tzafestas, "Robot fast adaptation to changes in human engagement during simulated dynamic social interaction with active exploration in parameterized reinforcement learning," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 10, no. 4, pp. 881–893, 2018.
- [21] G. Konidaris and A. Barto, "An adaptive robot motivational system," in *International conference on simulation of adaptive behavior*. Springer, 2006, pp. 346–356.
- [22] I. Cos, L. Canamero, G. M. Hayes, and A. Gillies, "Hedonic value: Enhancing adaptation for motivated agents," *Adaptive Behavior*, vol. 21, no. 6, pp. 465–483, 2013.
- [23] G. Baldassarre, "A planning modular neural-network robot for asynchronous multi-goal navigation tasks," in *Proceedings of the 2001 Fourth European Workshop on Advanced Mobile Robots-EUROBOT*, 2001, pp. 223–230.
- [24] S. Doncieux, N. Bredeche, L. L. Goff, B. Girard, A. Coninx, O. Sigaud, M. Khamassi, N. Díaz-Rodríguez, D. Filliat, T. Hospedales *et al.*, "Dream architecture: a developmental approach to open-ended learning in robotics," *arXiv preprint arXiv:2005.06223*, 2020.
- [25] G. Velentzas, C. S. Tzafestas, and M. Khamassi, "Memory development with heteroskedastic bayesian last layer probabilistic deep neural networks," in *Workshop on World Models and Predictive Coding in Cognitive Robotics at 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2023)*, 2023.