Affordance-Based Reasoning in Robot Task Planning

Iman Awaad and **Gerhard K. Kraetzschmar**
Bonn-Rhein-Sieg University
Grantham-Allee 20
53757 Sankt Augustin, Germany

Joachim Hertzberg
University of Osnabrück
49069 Osnabrück, Germany

Abstract

Humans are able to come up with plans to achieve their goals, and to adapt these plans to changes in their environment, finding fixes, alternatives and taking advantages of opportunities without much deliberation. For example, they may use a tea kettle to water the plants, or a mug instead of a glass. Despite decades of research, artificial agents are not as robust or as flexible. In this work, we introduce three reasoning phases that use affordances to enable such robustness and flexibility in robot task planning. The first phase generates a focused planning problem. The second phase expands the domain where necessary while the third and final reasoning phase uses affordances during plan execution and monitoring. This is accomplished by combining Hierarchical Task Network planning, description logics, and a robust execution/monitoring system.

Introduction

A paradigm shift has slowly been taking place among researchers in the various fields of AI, such as perception and manipulation (in the case of robotics). In recent years, we have seen an increasing number of approaches that are *task oriented*. For example, grasping objects is no longer solely dependent on the physical properties of the object to be grasped and that of the manipulator, but now often includes the purpose for which the grasping is taking place (for example, grasping an object to pour from it may require a different grasp than transporting it would).

In the planning field, the task-based perspective predates the current change in other robotics fields. The Hierarchical Task Network (HTN) approach (Ilghami and Nau 2003) has enabled us to reduce our search space by allowing us to encode the 'best way' of carrying out tasks; thus improving the quality of the plans as well. Complexity is at a minimum when there are no choices to be made - when there is exactly one way to decompose a non-primitive task into primitive tasks. This may however limit the possibility of generating a plan at all (if we have no way to decompose a non-primitive task that is applicable at a given state). What we would like to do is to make use of another behavior that humans often exhibit: we find 'other' ways to accomplish the task, quite often not so much by changing how we carry out the task but with what: we make 'substitutions'. For example, by using a mug instead of a glass to drink from or by using a tea kettle to water plants with instead of a watering can. This may be the equivalent of adding a new method to decompose the task that uses the substituted object (and depending on the result at execution, annotating it with a preference index) or in approaches that use lifting, it would enable additional objects to be used to ground the methods and operators. How would we determine what objects to substitute is the question.

In our approach, we will use a modified HTN planning algorithm that aims to re-use the procedural knowledge encoded in the methods while at the same time enabling the resourcefulness that humans exhibit when they substitute objects for other objects. To do this we use justification structures, as in (Veloso 1994; Fernandez and Veloso 2006), borrowed from explanation-based approaches to annotate the derivation process of a plan. This provides us with the ability to understand why a planner made a particular decision and why it may have failed to generate a plan. In the cases where this is due to a missing object, the algorithm uses a reasoning process that employs the concept of *affordances* to make effective choices.

Affordances describe opportunities for action (Gibson 1979). In this work, the notion of affordances is kept, although Gibson's action-perception coupling is not dealt with directly. Gibson's original definition has been refined by many researchers in numerous fields and a generally-agreed upon interpretation narrows the list of action-possibilities to the ones that an actor might be conscious of. Using the refined definition, affordances are neither solely a property of the object, nor of the actor, but of their relationship. We adopt Norman's definition (Norman 2002) (and the subsequent extensions of this definition by others, such as (Gaver 1991) and (Hartson 2003)) of "*perceived affordances*" which allude to "how an object may be interacted with based on *the actors's goals, plans, values, beliefs and past experience*" (Norman 2002). We propose to include affordances within the domain model and to represent this in Description Logics (DL) so that we may use the reasoning powers of existing tools to enable the robust and flexible behavior described above.

To this end, we attempt to answer the following specific questions: How does the agent recognize when it should make a substitution? How does it acquire the 'functional affordance' of an object? How is this functional affordance represented? When should it attempt to make a precondition true (for example, wash the dirty glass) as opposed to making a substitution (use a mug instead of the glass)?

The solution we propose incorporates ideas from each of these domains:

- Planning in highly dynamic domains where they need to consider actions which may or may not be possible due to changes in the state of the world. In our case, alternative actions and alternative objects also need to be considered in order to enable successful plan generation and execution. In our case, the choice is based on affordances, spatial proximity and preferences.
- Explanation-based approaches where information dealing with the decisions made by the planner are noted to help diagnose the planning process. Information such as the unavailability of objects with which variables can be grounded with can provide us with the necessary cue to expand the domain to include other types of objects. Unsatisfied preconditions and the affordance a method or operator was meant to enable would provide us with the information needed to make appropriate substitutions.
- Case-based planning where previous plans, or sub-plans, are used instead of new ones being generated. In our case in particular, with the 'creative' solutions which could result, it is desirable to keep track of both the plans that were considered good solutions as well as the ones that were not. Moreover, in any domain with tasks that are often repeated, such as making coffee every morning, it makes sense to avoid deriving a plan each time.
- Closely-related is planning by analogy where the similarity in problem descriptions identifies and enables a previous plan's structure to be adapted to the new case. Together with case-based planning and explanation-based planning, this approach provides us with a means to adapt the existing plans according to given guidelines.
- Opportunistic planning and reasoning approaches which focus on making use of opportunities to achieve goals that at a previous time were unachievable. This approach in particular seems well-suited to planning with affordances in mind as they would serve as the cues that trigger opportunistic behavior.

The goal of this work is to demonstrate the utility of using the powerful concept of affordances in the planning process to reduce complexity and increase flexibility - two tasks that may appear to be impossible to achieve simultaneously.

Application Scenarios

The scenarios presented below demonstrate how an affordance-based agent within the area of domestic service robotics would benefit from enhanced performance and increased robustness.

Object substitution A common household task involves fetching an object, for example, a glass. If one is not found during execution, an agent would have failed in achieving its goal. One equipped with affordance-based reasoning however, would use the affordances of a glass and substitute another item, such as a mug for it. In this case, the agent might use the functional affordances and/or physical properties of the objects to arrive at viable substitutions.

Action substitution The existence of an affordance depends as much on the morphology of the actor as it does on a particular set of features of the object. An object may be graspable for one embodiment but not for another. Even with the necessary morphology to allow grasping, a fault (be it temporary or not) with the system may prohibit the use of the manipulator. The object may also be too heavy to afford picking up. A human faced with a similar situation might attempt to push an object to its destination (assuming that its current and final positions allow this). Here, it is the similarity of the effects of the original action with its possible substitutes that has the greatest impact. The use of an action-ontology might help in enabling this use case.

Object substitution as tool usage Taking Norman's definition, the use of an actor's goals to pick up affordances can result in interesting uses of everyday objects. For example, the use of a magazine instead of a coaster for placing a bottle on a table is an affordance of a magazine that a human might take advantage of in order to achieve his/her goal of placing a cup on something other than the table. This use case illustrates this emergent behavior. Knowing the functional affordances of the original object and using the physical affordances of the substituted object would enable this use case.

Object substitution or use as performance enhancement Another common task involves fetching a cup of coffee, for example. If one uses affordances for planning, an agent could reason that a more appropriate object would be a mug for the coffee as opposed to the cup. In this case, the agent might use the functional affordances and/or physical properties of the objects to arrive at the most appropriate object.

From the use cases presented in the scenarios above, it is clear that modeling *functional affordances* are a necessity. These functional affordances can be seen as a subcategory of what Norman called "perceived affordances" which are based on experience and goals (not to be confused with the affordances which are perceived by the senses directly from the environment) . They are extremely important for a number of reasons. Firstly, they enable intelligent behavior by using objects for carrying out tasks that they were *meant* to. Secondly, the concept of affordances could very easily lead to an explosion in computational complexity as the action possibilities of objects are numerous (a chair can be sat on, but also thrown, stood on, etc.). By using functional affordances, the action space is successfully reduced (in the case when the substitution of objects is needed). In addition to functional affordances, *real affordances* which may be picked up from the environment through the perception process are also needed. A means to measure the similarity between objects would enable the synergistic use of these affordances for the object substitution, and object substitution as tool usage scenarios. For action substitution, the affordances are mainly related to the effects of the actions and a similarity measure between these would then robustly enable the scenario.

Generating the planning problem

The task of generating a problem description for planners is key. Part of this description is the domain. Modeling the domain is difficult and time consuming. The increasing complexity due to the size of the search space (caused by both the number of operators and the sheer number of objects in real world domains) remains a challenge - so much so that much of the benchmarking problems that are often used for planners can still be considered "toy problems" (Mastrogiovanni et al. 2010) .

Domain knowledge has long been used to help constrain the size of the planning problems. Hartanto takes this further by coupling DL reasoning with task planning. In (Hartanto 2009), he represents the domain in DL which enables him to infer a *constrained* planning domain by selecting only relevant elements (for example, only considering rooms whose doors are open in a navigation domain). The modeling of the domain is thus a crucial element in handling computational complexity. By linking affordances to tasks and representing these in DL we increase the use of domain knowledge and are thus able to improve the robustness of the system.

As mentioned above, complexity increases when a choice is necessary; when there is more than one way to accomplish a task or achieve a goal (when we have more flexibility). The generated plans' costs may differ greatly. The Hierarchical Task Network (HTN) planning approach improves the situation by providing an expert's way of carrying out the task; thus improving the quality of the plans. Moreover, in an environment shared by humans and artificial agents, this approach is beneficial as it is more human-readable and a good agent should be able to communicate their plan at all times (Bradshaw, Feltovich, and Johnson 2011). In addition, it enables the human user to specify the way he/she wishes to have a task accomplished in an intuitive way.

Let us take the task of watering plants as an example. The domain modeler would specify methods which provide both procedural knowledge (how the task is to be accomplished) and domain knowledge (specifying that a watering can in particular should be used) resulting in a task network such as that shown in Figure 1. If there is no watering can in our domain (or despite all of our methods and operators no decomposition is found to accomplish the task), the plan generation process will fail (for example, the watering can exists but is inaccessible and we have no means by which to make it accessible).

In our approach, we propose a modified HTN planning algorithm that aims to re-use the procedural knowledge of the methods while at the same time exhibiting the resourcefulness that humans show when they substitute objects for other objects. First, we need to answer the question of how the agent recognizes when it should make a substitution. To do this we will use justification structures, as in (Veloso 1994; Fernandez and Veloso 2006), borrowed from explanation-based approaches to annotate the derivation process of a plan. This provides us with the ability to understand why a planner made a particular decision and why it may have failed to generate a plan. In the cases where this is due to a missing object, the algorithm uses a reasoning process that employs the concept of *affordances* to expand the domain accordingly to make effective choices (this might be the most appropriate, or the cheapest choice).

Expanding the domain

In (Magnenat, Chappelier, and Mondada 2012), the authors use the HTN domain to constrain the search space and then learn the probabilities of success in order to enable more robust plans. The lifting process is done over categories of objects rather than instances, thus reducing the complexity. The proposed use of this lifting over categories of objects and/or their functional affordances in our approach provides us with the chance to increase the number of possible objects to be ground and at the same time remain focused on achieving the task by choosing a more general category or a functional affordance.

The question is, how does the robot decide which substitutions are admissible? We argue that the most appropriate substitutions are the ones that are meant to be used for the same task. For example, glasses and mugs are both used to drink from. This is knowledge that humans learn and that may be found in the dictionary for example, or possibly through projects that aim to make common-sense knowledge available to artificial agents (such as OpenCyc (Cycorp 2013), RoboEarth (Hubel et al. 2010), ConceptNet and WordNet). These knowledge sources may be used to provide Norman's (Norman 2002)"values, beliefs" and even "past experience". Objects that are used for the same task (i.e. share the same 'functional affordance') would provide the most 'appropriate' substitutions. In addition, they can of course also come from the humans co-inhabiting the environment (for example, they might ask the robot to only clean the bathrooms with the blue cleaning cloths (restricting it to the subcategory), or to only serve them tea in their favorite cup (a single instance)). This answers the question of where the functional affordances of objects come from.

Some objects, such as watering cans, are used for a very specific task. In this example, for watering plants. The only other object which is used for the same task would be a 'hose', and this is only for watering plants outdoors. In this case, both share the same functional affordance of watering plants, but whereas it may be desirable to substitute the watering can for the hose, the opposite is not true, and so a substitution using only functional affordances may fail. Here, we would need to look for objects which are conceptually similar to the watering can. The similarity measures which are often used may not yield the results we have in mind (we may not care about the color of an object, but rather the presence of a handle for example).

We propose the use of *Conceptual Spaces* (Gärdenfors 2004) which provide a multi-dimensional feature space (each base or axis is referred to as a quality dimension, for

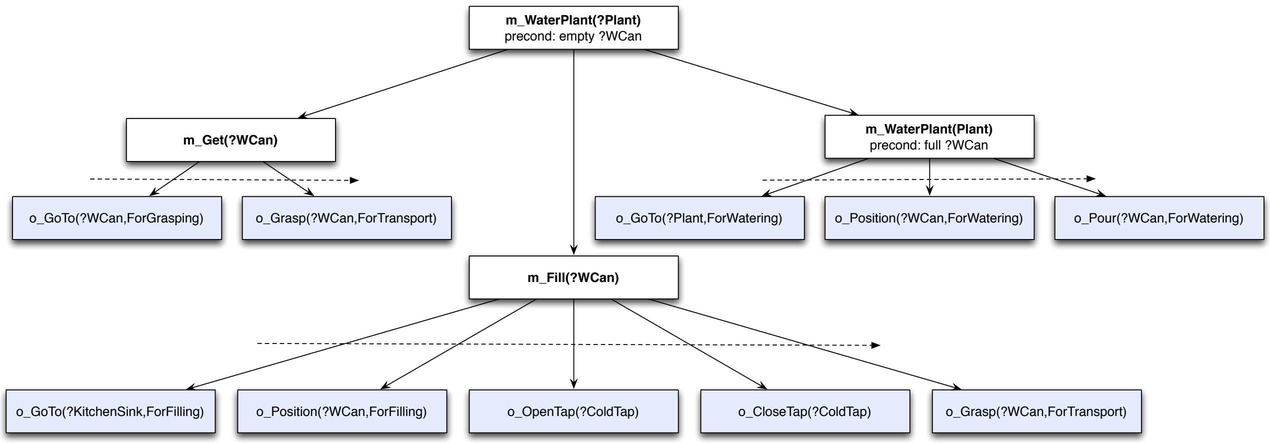

Figure 1: Task network for the WaterPlant method

example color, or height) that can be used to describe conceptual similarity. Here, points in a space would represent objects, and regions would refer to concepts. As conceptual spaces provide various quality dimensions (some or all of which may be sensed by the agent, depending on their sensing capabilities), the idea is to see if we can determine a relation between these quality dimensions and given tasks (for example, for the task of lifting an object, the most important quality dimension would be its weight - its color would be irrelevant). These relations could then be used as weighting factors to determine how good an object would be as a substitute for another in achieving a given task. Conceptual spaces can also be used to represent shape, such as handles, or spouts for example. The detection of these quality dimensions would obviously require more processing by the perception components than the simple detection of hue for example. This would serve as a more robust and 'focused' similarity measure for achieving a given task. In the case of finding a substitute for the watering can, and given that for such a task, the capacity to hold water is perhaps the most important affordance, followed by the presence of a handle and a spout, conceptual spaces could find that the most similar item would be the tea kettle.

If we are to make use of the reasoning power of DL, it would be beneficial to represent the Conceptual Spaces (with the information they hold) in DL. There exists already the *Conceptual Spaces Markup Language* (Adams and Raubal 2009). The use of this standard and the ability to reason with it will be investigated.

Plan execution and monitoring

Having successfully provided a compact planning problem to the planner and generated a plan, its execution and careful monitoring thereof is necessary. During this phase, a number of issues need to be addressed. We would like the system to robustly handle unexpected situations and to take advantages of opportunities.

Unexpected situations could occur due to partial observability of the environment or as a result of a dynamic environment (for example, a door which was previously known to be open may be closed at the time of execution, or the watering can which was known to be in a given location can no longer be found). In this case, the system behaves much as it might during the plan generation phase, with a slight difference.

Just as humans prefer to take advantage of objects within their immediate spatial surroundings in such situations, the agent might do the same. In the example of the coaster and the cup presented in the 'application scenarios' section above, humans would no doubt consider the use of objects which are already on the table in the absence of the coasters (such as magazines). Similarly, through the use of affordances, we hope to accomplish the same. In order to truly take advantages of opportunities within the environment, which by definition are unexpected opportunities, we need to combine both the execution of plans which have been generated through the deliberation process and reactive behaviors which may be triggered by affordance cues.

We propose a simple blackboard architecture where affordance cues (in the form of conceptual space quality dimensions) are being posted as the agent moves through its environment as part of executing a plan. These might be of varying complexity (from simple color hues which would cost very little in terms of perceptual processing to more complex concepts such as shape which might have been picked up as part of the plan's execution) and would be kept in the system for a given duration. Upon plan failure, the cues which are in close proximity can be used to identify viable candidates for substitutions.

Of course, the same behavior can be used to guide plan execution even when things are going as planned and of course to take advantage of opportunities before failures occur. For example, cues that are associated with a drink bottle may have been picked up on the way to the location specified in a plan. This 'short cut', could be taken advantage of. A cupboard full of glasses would guide the agent to grasp any of them (if there are no additional constraints like using a specific cup for example); and in the case of plan failure, and depending on the desired behavior, an agent might take the more 'resourceful route' of making a substitution or attempt to use the same object by finding other instances or using objects with the same functional affordance. For action substitution, the case is very much the same but is mostly applicable in the case of failures in plan generation or plan execution.

Work plan

Since May 2012, I have completed a review on related work in affordance-representation and closely-related approaches in automated planning. I have developed the architecture of the system and its integration into the current framework of our b-it-bots RoboCup@Home team robot. I've identified, tested and chosen the tools and libraries that I will use in the system and am continuing to work through the design and implementation of the first use case. This involves modeling the functional affordances in DL. The object substitution use case will be used to validate the designed model of the functional affordances. Extending JSHOP2 to use lifting over categories and the justification structures is also a current task. Designing the plan library (including preferences, etc.) will be the next major task.

Having completed and tested the plan generation and domain expansion phases of the approach using functional affordances, I will extend this to enable the second use case: action substitution. The instantiation of affordance behaviors at execution time using conceptual spaces will be investigated. With these first two use cases complete, the third use case of object substitution as tool usage is then an extension of the work as both the functional affordances and the use of conceptual spaces for representing the affordance cues and conceptual similarity necessary to enable this use case will have been integrated and validated. The final use case for enhancing performance by specifying more general plans, or identifying 'better' ways to accomplish a task will be investigated. In the process of enabling the previous use case, it is expected that a natural abstraction hierarchy will have emerged. Throughout, it will be necessary to enhance and extend the plan library and its use to recognize, and adapt existing solutions to new problems.

References

Adams, B., and Raubal, M. 2009. Conceptual Space Markup Language (CSML): Towards the Cognitive Semantic Web. In *2009 IEEE International Conference on Semantic Computing*, volume 0, 253–260. Los Alamitos, CA, USA: IEEE.

Bradshaw, J. M.; Feltovich, P. J.; and Johnson, M. 2011. *The handbook of human-machine interaction: a human-centered design approach*. Farnham, Surrey, England; Burlington, VT: Ashgate. chapter 13, 283–300.

Cycorp. 2013. Opencyc. Online at www.opencyc.org.

Fernandez, F., and Veloso, M. 2006. Probabilistic policy reuse in a reinforcement learning agent. In *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multi-Agent Systems*.

Gärdenfors, P. 2004. Representing actions and functional properties in conceptual spaces. *to appear: Body, Language and Mind, T. Ziemke and J. Zlatev, Editors*.

Gaver, W. W. 1991. Technology affordances. In *Proceedings of the SIGCHI conference on Human factors in computing systems: Reaching through technology*, CHI '91, 79–84. New York, NY, USA: ACM.

Gibson, J. J. 1979. *The ecological approach to visual perception*. Houghton Mifflin (Boston).

Hartanto, R. 2009. *Fusing DL Reasoning with HTN Planning as a Deliberative Layer in Mobile Robots*. Ph.D. Dissertation, University of Osnabrück.

Hartson, H. R. 2003. Cognitive, physical, sensory, and functional affordances in interaction design. *Behaviour & IT* 22(5):315–338.

Hubel, N.; Mohanarajah, G.; van de Molengraft, R.; Waibel, M.; and D'Andrea, R. 2010. Roboearth project. Online at http://www.RoboEarth.org.

Ilghami, O., and Nau, D. S. 2003. A General Approach to Synthesize Problem-Specific Planners. Technical Report CS-TR-4597, UMIACS-TR-2004-40, University of Maryland.

Magnenat, S.; Chappelier, J.-C.; and Mondada, F. 2012. Integration of Online Learning into HTN Planning for Robotic Tasks. In *Proceedings of the AAAI Spring Symposium 2012: Designing Intelligent Robots, Reintegrating AI.*

Mastrogiovanni, F.; Scalmato, A.; Sgorbissa, A.; and Zaccaria, R. 2010. Affordance-based planning for assisting humans in daily activities. In *Proceedings of the 2010 Sixth International Conference on Intelligent Environments*, IE '10, 19–24. Washington, DC, USA: IEEE Computer Society.

Norman, D. 2002. *The psychology of everyday things*. Basic Books (New York).

Veloso, M. M. 1994. Flexible strategy learning: Analogical replay of problem solving episodes. In *Proceedings of AAAI-94, the Twelfth National Conference on Artificial Intelligence*, 595–600. Seattle, WA: AAAI Press.