



Taskability Graph: Towards Analyzing Effort based Agent-Agent Affordances

Amit Kumar Pandey, Rachid Alami

► To cite this version:

Amit Kumar Pandey, Rachid Alami. Taskability Graph: Towards Analyzing Effort based Agent-Agent Affordances. IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication, Sep 2012, Paris, France. hal-01977526

HAL Id: hal-01977526

<https://laas.hal.science/hal-01977526>

Submitted on 10 Jan 2019

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.

Taskability Graph: Towards Analyzing Effort based Agent-Agent Affordances

Amit Kumar Pandey and Rachid Alami

Abstract—Affordance analysis, what something/someone can afford or offers, is an important aspect for day-to-day interaction and decision-making. In this paper, we will enrich the notion of affordance by incorporating *agent-agent affordance*: what does an agent afford for another agent in terms of a task. Further, we will present an effort hierarchy and derive the concept of *Taskability Graph*, which encodes: *what all agents could do for all other agents, with which levels of mutual-efforts and at which places*. This makes the robot more aware about agents’ abilities and facilitates to develop better interaction and decision-making capabilities. We will discuss the potential application in effort based shared cooperative planning.

I. INTRODUCTION

In cognitive psychology, Gibson [1] refers affordance as what an object offers. He defines affordances as all action possibilities, independent of the agent’s ability to recognize them. Whereas in Human Computer Interaction (HCI) domain, Norman [2] tightly couples affordances with past knowledge and experience and sees affordance as perceived and actual properties of the things. In robotics affordance has been viewed from different perspectives: agent, observer and environment, [3]. Irrespective of shift of definition, affordance has been regarded as an important aspect for a socially situated agent for day-to-day interaction and decision-making. Affordance itself could be learnt [4] as well as could be used to learn action selection [5].

Affordance has been studied in robotics with respect to object (e.g. for tool use [6]) and location (e.g. for traversability [7]). However, rich geometric reasoning based *what* an agent affords to do for another agent (give, show, hide, make accessible, ...) has not been explored from human robot interactive task point of view. In this paper, we will address this aspect by introducing the complementary notion of *Agent-Agent Affordance*. The robot will proactively compute *agent-agent affordances* for all the agents and for a set of basic human-robot interactive tasks. This will be encoded in a graph, which we termed as *Taskability Graph*. This will enable the robot to maintain in real time what an agent can afford to do for another agent, with which effort level and at which places. This will facilitate a more ‘aware’ human-robot interaction.

In the next section, sub-section II-A will briefly and formally present the concept of *Mightability Maps*, which in

fact will serve as the basis for finding *agent-agent affordance*. Sub-section II-B will present *Human-Aware Effort Hierarchy*, to ground the agent’s effort in a human-understandable form. Sub-sections II-C to II-E will present the approach to find the affordance of a task for a particular agent-agent mutual-effort criterion. Subsequently we will develop the concept of *Taskability Graph*. Experimental results of section III will be followed by discussion on potential applications and conclusion with pointer to the future work.

II. METHODOLOGY

A. Mightability Maps: Multi-state visuo-spatial perspective taking

Visuo-spatial perspective-taking has already been shown as an important component in Human-Robot Interaction, [8], [9]. We have enriched the robot’s ability of perspective taking not only from the current state of the agent but also from a set of different states, the agent might attain. Based on this, in [10], we have presented the concept of *Mightability Maps*. *Mightability* stands for “*might be able to...*” and facilitates multi-state visuo-spatial perspective taking. The idea is to analyze the various abilities, A_b , of an agent by applying an ordered list of virtual actions, $A_V = [a_1, a_2, \dots, a_n]$. Respecting the environmental and postural constraints, the robot performs Mightability Analysis by taking into account collision as well as the joint limits. When the robot estimates the abilities at 3D grid level, we termed it as *Mightability Maps (MM)*, [10]. In the current implementation A_b :

$$A_b \in \{to\ see, to\ reach\} \quad (1)$$

$$a_i \subseteq \{A_V^{head}, A_V^{arm}, A_V^{torso}, A_V^{posture}, A_V^{displace}\} \quad (2)$$

where,

$$A_V^{head} \subseteq \{Pan_Head, Tilt_Head\} \quad (3)$$

$$A_V^{arm} \subseteq \{Stretch_Out_Arm\ (left|right)\} \quad (4)$$

$$A_V^{torso} \subseteq \{Turn_Torso, Lean_Torso\} \quad (5)$$

$$A_V^{posture} \subseteq \{Make_Standing, Make_Sitting\} \quad (6)$$

$$A_V^{displace} \subseteq \{Move_To(position)\} \quad (7)$$

These *Mightability Maps* will be used to find candidate places for agent-agent affordance in section II-C.

B. Human-Aware Effort Hierarchy (HAEH)

For a robot to interact and cooperate with us in complete harmony, it should be able to reason on effort at the human-understandable level of abstraction. We have conceptualized an effort hierarchy based on the body parts involved in

This work has been conducted within the EU SAPHARI project (<http://www.saphari.eu/>) funded by the E.C. Division FP7-IST under Contract ICT-287513.

Authors are with CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France; Univ de Toulouse, LAAS ; F-31400 Toulouse, France
akpandey@laas.fr; rachid.alami@laas.fr



Fig. 1: Taxonomy of Reach Types.

Effort to Reach	Effort Level	Effort to See
No_Effort_Required	<div style="display: flex; align-items: center; justify-content: center;"> <div style="width: 10px; height: 100px; background: linear-gradient(to bottom, green, yellow, orange, red); border: 1px solid black; margin-right: 5px;"></div> <div style="text-align: center;"> <p>Minimum:0</p> <p style="color: red; font-weight: bold;">↓</p> <p>Maximum:5</p> </div> </div>	No_Effort_Required
Arm_Effort		Head_Effort
Arm_Torso_Effort		Head_Torso_Effort
Whole_Body_Effort		Whole_Body_Effort
Displacement_Effort		Displacement_Effort
No_Possible_Known_Effort		No_Possible_Known_Effort

Fig. 2: Human-Aware Effort Hierarchy: Qualifying efforts to see and to reach an object or a place.

performing an action. This is motivated from the studies of human movement and behavioral psychology, [11], [12], where different types of reach actions of the human have been identified and analyzed, such as, reach involving simple arm extension (arm-only reach), shoulder extension (arm-and-shoulder reach), leaning forward (arm-and-torso reach) and standing, see fig. 1. We also associate effort level to a particular A_v , based on the joints involved, as follows:

$$E_{see} \in \{No_Effort, Head_Effort, Head_Torso_Effort, Whole_Body_Effort, Displacement_Effort\} \quad (8)$$

$$E_{reach} \in \{No_Effort, Arm_Effort, Arm_Torso_Effort, Whole_Body_Effort, Displacement_Effort\} \quad (9)$$

Table in fig. 2 summarizes this *Human-Aware Effort Hierarchy (HAEH)*. To illustrate, let us assume that an agent Ag is currently sitting. From this current state, Ag can put different efforts to attain different states to see or reach something or to perform some task. If Ag only turns head to see an object or place, we term it as *Head_Effort*. If Ag has to turn the torso also, it is *Head_Torso_Effort*. *Whole_Body_Effort* involves additional joints, instead of just turning the head or the torso, e.g. if $A_v = [Make_Standing, Lean_Torso]$. *Displacement_Effort* corresponds to the situations requiring Ag to move from the current position. Similarly, if Ag only stretches out the arm (to point, to reach,...), it is *Arm_Effort*. If Ag has to also turn around or lean, it is *Arm_Torso_Effort*, and so on. Hence, this effort hierarchy grounds the agent's movement to a 'meaningful' effort. The robot further associates descriptors like left and right.

Apart from facilitating better verbal interaction, *HAEH* also facilitates the robot to 'understand' and incorporate different high-level constraints related to the desire and physical state of Ag , in decision making and cooperative task planning. E.g. if Ag is suffering from back or neck pain, or facing challenges in standing up or has reduced mobility, these can be directly incorporated in terms of high-level constraints, by instructing the planner to exclude the efforts associated with the torso or head movement or by restricting maximum effort as torso effort, respectively.

Similarly current situation, preferences and mutual responsibilities could also be grounded in terms of these effort

levels. For example if Ag is tired, his/her/its effort level can be restricted to a particular level, e.g. *Torso_Effort*. *HAEH* also facilitates 'meaningful' reasoning on mutual effort, which will be discussed in section II-D.

Within each these effort levels, further a quantitative value is assigned based on how much the body part has to be moved or turned. This could be further enriched based on musculoskeletal kinematics and dynamics models [13], [14].

C. Finding task based candidate places

Currently the robot is equipped to analyze the basic Human-Robot Interactive manipulation tasks: *Give, Show, Hide, Make-Accessible*. For finding the candidate places for a particular task, a set of constraints $Cnts$ is used:

$$Cnts = \{c_i : i = 1 \dots m\} \quad (10)$$

m is total number of constraints and c_i consists of tuple:

$$c_i = \langle agent(A_g), effort(E), ability(A_b) = true|false \rangle \quad (11)$$

Where A_b is element of eq. 1, E is element of eq. 8 or 9. Depending upon the task, the desired values as *true* or *false* are known a priori for a particular ability. For example if the task is to *give*, the planner knows that the abilities to *see* and *reach* the candidate places should be *true* for both the agents. Based on these constraints, the planner performs set operations on the *Mightability Maps*, [10], and obtains the set of candidate points as:

$$P_{place}^{obj, Cnts} = \{p_j : j = 1 \dots n \wedge p_j \equiv (x, y, z) \wedge (p_j \text{ satisfies } \forall c_i \in Cnts)\} \quad (12)$$

Where n is the number of places. Eq. 12 consists of commonly reachable and visible places for hand-over task, places to put object for hide task, etc. with particular effort levels of the agents. If *obj*, the object for which the task is to be performed, is provided beforehand, depending upon the nature of the task, the *Mightability Maps* are grown or shrunk based on the object dimension. This is to include or exclude the possible candidate places because of the object size. We assume that the robot is provided with constraints $Cnts$ for the set of basic tasks to find the candidate places, of eq. 12. However, it can also be learnt in terms of desired effect [15].

D. Reasoning on Mutual-Efforts

As long as the robot reasons only based on the current states of the agents, the complexity as well as the flexibility of cooperative task planner is bounded. For example, if an agent cannot reach an object or cannot give an object from the current state, it means, to the planner, the agent will not be able to manipulate that object or will not be able to perform hand-over task at all. But if the robot is aware about the agent's ability to do something with different effort levels, this provides latitude to better solve for the task but introduces another dimension of '*deciding effort*' in the cooperative task planning. As theoretically, the robot can find



Fig. 3: Illustration of different mutual efforts. (a) Initial Scenario. Task: P1 has to make-accessible the green bottle (indicated by red arrow) to P2. (b) *Reducing Self Effort*: P2 has to stand up and lean forward to take the bottle. (c) *Reducing Other's Effort*: P1 stands up and puts it at a place so that P2 can take it just by stretching out her arm. (d) *Effort Balancing*: P1 puts by leaning forward while remain seated, so that P2 will also be required to lean forward to take.

out that each agent might be able to perform the tasks, only the effort to do so will vary.

For example, consider scenario of fig. 3, in which the green bottle (marked by arrow) is currently not visible and reachable to P2, but to P1. The task is: *Performing-agent P1* has to make the bottle accessible to P2, the *target-agent*. As shown in the figure, different mutual-effort decisions result into different placements of the object: reducing self-effort in fig. 3(b), reducing other's effort in fig. 3(c) and balancing mutual efforts in fig. 3(d).

Our framework facilitates reasoning on different mutual-effort criteria, which could be passed in eq. 11 to find affordances as will be discussed in the next section. Below we will discuss different decisional aspects of mutual-effort, which could be easily incorporated in our framework.

As being in human-centered environment, the robot will be expected to help, cooperate, collaborate, and find shared plans to distribute the workload and balance/reduce the efforts. If the agents share equal levels of responsibility, they will be expected to be *mutually responsive* and *mutually responsible*, hence should try to *balance mutual effort*.

The desired effort level could also be restricted based on the current context. For example, if the agents are sitting around a table, they may wish to avoid displacement effort to take something; instead might request to someone else who can easily take and hand-over the object. Hence, trying to reduce self-effort. If the role of the robot is as a caregiver to some elder person, there will be a need to reduce the person's effort at the cost of self-effort.

In summary, various global and individual factors influence the effort level of an agent: *social status*, *responsibility level*, *current context* (dining, reception, living room), *in-*

dividual desire (not willing to move), state (tired, reduced mobility, back or neck problem, aged,...), *role* (caregiver, boss, friend), etc.

E. Finding mutual-effort based Agent-Agent Affordances

For finding the agent-agent affordance with the criterion of balancing mutual-effort, the approach is as follows: The planner sets the initial effort levels for both the agents as least effort in the set of constraints of eq. 10, which is *No_Effort_Required* to reach and to see (refer table of fig. 2). Then the planner obtains in eq. 12 the set of candidate places for the task. If the resultant candidate set is NULL then the planner sequentially increases the efforts of both the agents to the next level until a NOT NULL candidate space is obtained or the maximum allowed effort levels of the agents are reached.

By regulating the agents' effort levels, different criteria of mutual efforts are incorporated. For example, to minimize the *target-agent's* effort, in the iteration, only the effort of the *performing-agent* is increased while maintaining the target agent's effort as lowest feasible. Opposite is done if the *performing-agent's* effort has to be minimized.

In the current implementation, as first level of estimation, we assume if the algorithm results into a NOT NULL candidate space, the agent might afford to perform the task. This serves three important purposes: (i) Since the affordances are calculated based on *Mightability Maps*, which are updated online (see [10]), the robot is aware about the affordances almost in real time. (ii) There is no false negative, i.e. it ensures that if an agent cannot afford a task, he/she/it indeed will not be able to, for the allowed effort level. (iii) Further, it ensures that there will be no solution outside

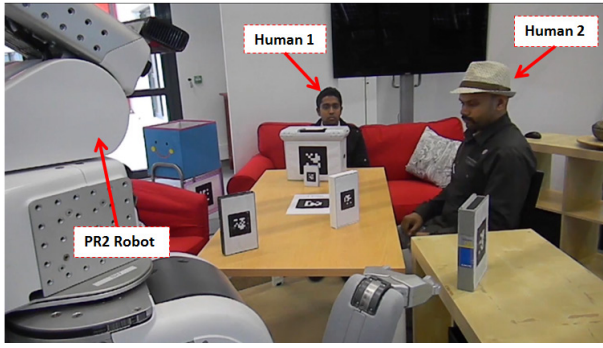


Fig. 4: Scenario of a typical human-robot interaction.

the resultant candidate set. We have presented in [16], a constraint hierarchy based approach for planning Human-Robot Interactive Object Manipulation tasks. It takes into account various constraints from the perspectives of the task, the environment and the human, to find the feasibility of a task. Hence if required, this can be used to perform a more precise analysis of agent-agent affordance to construct more precise *Taskability Graph* (discussed in next section). However, we prefer to delay such computationally expensive constraints and feasibility tests until a task has been decided to be performed, this is mainly to maintain a rough but almost online estimation of affordances.

F. Taskability Graph

Taskability Graph encodes what each agent might be able to do for every other agent, with which levels of effort for both the agents, and at which places.

For constructing *Taskability Graph* the criterion of mutual effort (effort-balancing, reducing self-effort, reducing other's effort,...) is provided. It is constructed by finding individual pairs of agent-agent affordances as discussed in previous section among all the agents. Currently 4 tasks are encoded in the *Taskability Graphs*: *Make Accessible*, *Show*, *Give* and *Hide*. The allowed maximum effort levels (based on situation, state, desire, ...) of each agent can also be provided at the time of constructing the *Taskability Graph* or could be used at later stages to filter/refine the graph.

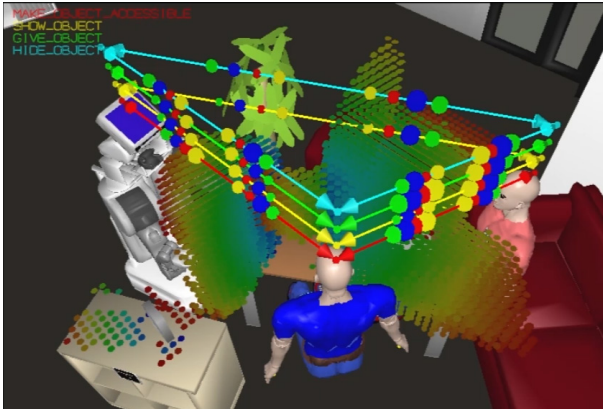


Fig. 5: *Taskability Graphs* for different tasks based on criterion of effort balancing, assuming the agents share equal responsibility. Another criterion was provided as the desired maximum individual effort level as *Arm_Torso_Effort*.

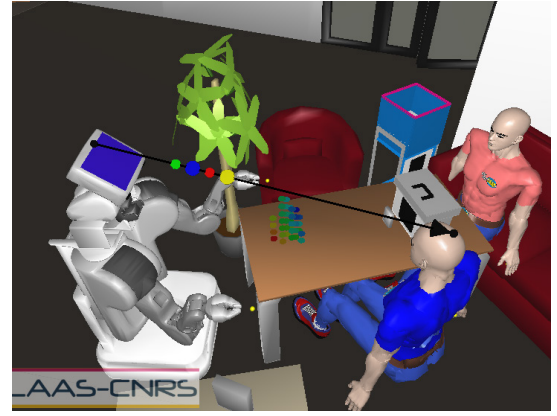
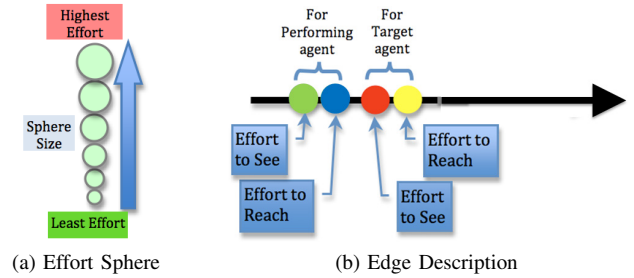


Fig. 6: Example of an edge of a *Taskability Graph*



(a) Effort Sphere

(b) Edge Description

Fig. 7: Explanation of an edge of a *Taskability Graph*.

III. EXPERIMENTAL RESULTS AND ANALYSIS

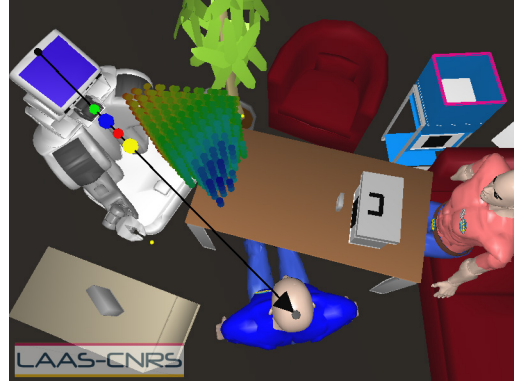
We have experimented with real robot PR2 in various human-robot interaction (HRI) scenarios. Fig. 4 shows one of such typical HRI scenarios: the PR2 robot and two human are sitting around a table. The robot, through various sensors, maintains and updates the 3D world state in real time. For this, it uses Move3D, [17], an integrated planning and 3D visualization platform. For object identification and localization, tags based stereovision system is used. For localizing the human and tracking the whole body, data from *Kinect* (Microsoft) sensor is used.

Fig. 5 shows the *Taskability Graphs* for different tasks for the scenario of fig. 4. The 4 layers of edges, in bottom up order, show affordances for *make accessible*, *show*, *give* and *hide* tasks among the agents.

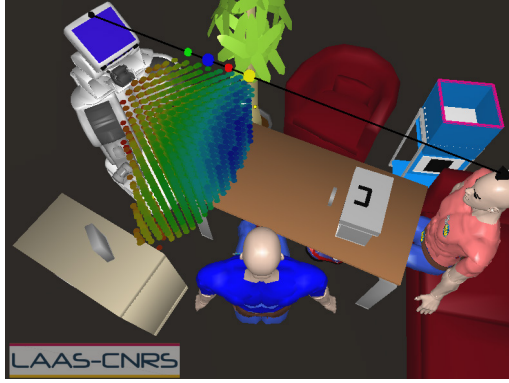
Fig. 6 shows an edge of the *Taskability Graph*. This corresponds to PR2's affordance of *give* task to the human 2. Fig. 7 explains what does an edge reveal. It is a directed edge from *performing agent* (source vertex) to *target agent* (target vertex). On the first half (from source to target vertex) of each edge there will be 4 spheres, showing the required effort levels of both the agents to perform the task. If an edge is bidirectional, it means both the agents could perform that task for each other. In that case, there will be 8 spheres on the edge for two different affordances. Associated with each edge there is also a set of weighted candidate places, indicating, where the task could potentially be performed with these effort levels (weights are assigned similar to [10]). The colored point cloud between the agents in fig. 6 shows the candidate places for the *give* task. On the edge of fig. 6, the small size of green and red spheres represents *No_Effort*,



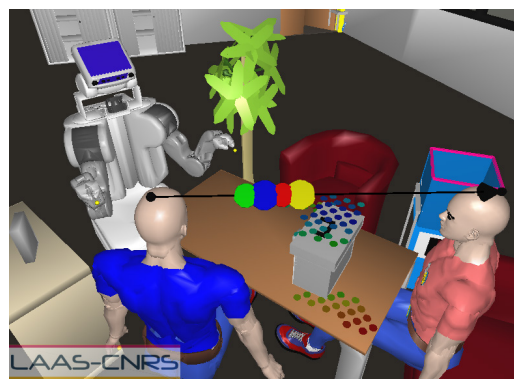
(a) PR2 affordance of *hide* task for human 1



(b) PR2 affordance of *show* task for human 2



(c) PR2 affordance of *show* task for human 1



(d) Human 2 affordance of *make-accessible* task for human 1

Fig. 8: Subsets of edges of *Taskability Graph* of fig. 5, encoding different *agent-agent affordances*.

which encodes that the hand-over places are in the current field of view of both the agents. Whereas, the relative bigger size of blue and yellow spheres shows that the agents will be required to put at-least *Arm_Effort* to reach the places to perform the task.

For constructing the *Taskability Graph* of fig. 5, the criterion was to balance the mutual-effort. Therefore, the corresponding spheres for *performing* and *target* agents in each edge are roughly equal in size. Further, the current context of sitting around a table has been also used to restrict the individual maximum desired effort as *Arm_Torso_Effort*. Hence, between the human on the right and the robot there is no possibility of *give* and *make accessible* tasks as reflected from the missing edges between these two agents in the *Taskability Graph* of fig. 5. It is also interesting to note the different size of effort spheres for different tasks among different agents. Fig. 8 shows a subset of different agent-agent affordances encoded in the *taskability graph* of fig. 5. Interestingly in fig. 8(a), which show *hide* task affordance of PR2 for *human 1*, the places to hide an object has been found behind the cassette on the shelf as well as inside the shelf. Fig. 8 (b) and (c), encode *show* task affordance of PR2 for different humans. Each has different candidate places. This is because of provided mutual-effort balancing criterion, the planner tries to assure least feasible effort of each agent. Hence, the places, which are reachable and visible by least feasible effort of PR2 and also visible by the

target-agent's least feasible effort are different, for different relative positions of the two target-agents. Fig. 8(d) shows the affordance of *human 2* for *make-accessible* task. The candidate places are autonomously found on top of table as well as on the box.

As the affordances are calculated based on *Mightability Maps*, which are updated online ([10]), *Taskability Graph* could also be updated almost in real time. In the current implementation, given the *Mightability Maps*, it takes about *1.2 seconds* for computing the *Taskability Graph* of fig. 5. Although this delay is acceptable for a typical human-robot interaction, it will benefit from the algorithms to intelligently update only the required parts of the already computed *Taskability Graph*.

IV. DISCUSSION AND POTENTIAL APPLICATIONS

Taskability Graph makes the robot aware about the capabilities of all the agents in the environment for different tasks they could perform. This serves one important purpose of planning cooperatively for a task by taking into account the effort. As it is a graph, by assigning proper weights to the edges, any graph search algorithm can be used to find a cooperative solution for the task. For example, consider that PR2 robot has to give some object to *human 1* of fig. 4. It can easily find "how" to give, by finding the shortest path from the start vertex (SV) corresponds to PR2 and the target vertex (TV) corresponds to *human 1* in the corresponding *Taskability Graph* of fig. 5. In the current



Fig. 9: Execution of *Taskability Graph* based generated shared plan to give an object to *human 1* of fig. 4 by planned help from *human 2*, sitting in the middle.

example, the resultant shared plan to give the object, obtained through the shortest path is: $PR2(\text{Give}, \text{to}(\text{Human2})) \rightarrow \text{Human2}(\text{Take}, \text{from}(PR2)) \rightarrow \text{Human2}(\text{Give}, \text{to}(\text{Human1})) \rightarrow \text{Human1}(\text{Take}, \text{from}(\text{Human2}))$.

Note that the resultant plan autonomously takes the help of *human 2*. The plan also globally minimizes the overall effort, because it corresponds to the shortest path in the *Taskability Graph*, and the weights of the edges are based on the efforts. Fig. 9 shows the execution of this plan, where the robot gives the object to *human 2* and he gives it to *target-human 1*.

It is interesting to note that by adjusting the weights of the edges or by excluding/including a particular vertex, different constraints could be further incorporated dynamically. For example, to find a plan, which minimizes effort of a particular agent or excludes/involves a particular agent, etc.

By reasoning on the *Taskability Graphs* of different tasks, the robot could even plan to cooperate with one agent to compete with other. E.g., the robot could plan to ask for an object from *human 2*, so that it could hide it from *human 1*.

Taskability Graph can also enrich verbalized human-robot interactions as well as could facilitate grounding the agents and actions for a change in the environment. For example if a previously visible object is currently not visible to the robot, then based on *Taskability Graph*, the robot could 'guess': which agent could afford to hide the object from the robot perspective and with which effort level. It could also reason about any potential cooperation among the agents as well as the potential locations to search for that object.

V. CONCLUSION AND FUTURE WORK

In this paper, we have presented the notions of *agent-agent affordances* and *human-aware effort hierarchy* and derived the concept of *Taskability Graph*. *Taskability Graph* encodes rich information about *agent-agent affordances*, which makes

the robot aware about what an agent can afford to do for another agent, with which levels of mutual-effort and at which places. We have discussed that *Taskability Graph* could be used to find shared cooperative plan by incorporating various effort based requirements and to ground agents and actions. Computation of *Taskability Graph* is almost online and could facilitate smooth and better Human-Robot Interaction.

One complementary aspect, from the perspective of human-robot interactive object manipulation tasks, is to analyze *agent-object affordances*. That is to build a similar graph based on what an agent can do with an object, by similar reasoning of efforts. Merging both graphs will make the robot 'aware' about: what an agent can manipulate and what he/she/it can do for another agent and with which effort level. This will facilitate incorporating the objects in various decision making and planning aspects, within the graph search framework presented in this paper.

REFERENCES

- [1] J. J. Gibson, *The Theory of Affordances*. Psychology Press, Sep. 1986, pp. 127–143.
- [2] D. Norman, *The psychology of everyday things*. Basic Books, 1988.
- [3] E. Şahin, M. Çakmak, M. R. Doğar, E. Uğur, and G. Üçoluk, "To afford or not to afford: A new formalization of affordances toward affordance-based robot control," *Adaptive Behavior - Animals, Animats, Software Agents, Robots, Adaptive Systems*, vol. 15, no. 4, pp. 447–472, Dec. 2007.
- [4] E. J. Gibson, "Perceptual learning in development: Some basic concepts," *Ecological Psychology*, vol. 12, no. 4, pp. 295–302, 2000.
- [5] M. Lopes, F. S. Melo, and L. Montesano, "Affordance-based imitation learning in robots," in *IROS*, 2007, pp. 1015–1021.
- [6] A. Stoytchev, "Behavior-grounded representation of tool affordances," in *ICRA*, April 2005, pp. 3060 – 3065.
- [7] E. Uğur, M. R. Dogar, M. Cakmak, and E. Sahin, "Curiosity-driven learning of traversability affordance on a mobile robot," in *IEEE 6th International Conference on Development and Learning (ICDL' 07)*, July 2007, pp. 13–18.
- [8] M. Berlin, J. Gray, A. I. Thomaz, and C. Breazeal, "Perspective taking: an organizing principle for learning in human-robot interaction," in *AAAI*, vol. 2, 2006, pp. 1444–1450.
- [9] S. C. Eimler, N. C. Krmer, and A. M. vob der Pten, "Prerequisites for human-agent- and human-robot interaction: Towards an integrated theory," in *European Meetings on Cybernetics and Systems Research (EMCSR)*, edited by R. Trappl. Vienna: Austria, 2010.
- [10] A. K. Pandey and R. Alami, "Mightability maps: A perceptual level decisional framework for co-operative and competitive human-robot interaction," in *IROS*, 2010, pp. 5842 – 5848.
- [11] H. J. Choi and L. S. Mark, "Scaling affordances for human reach actions," *Human Movement Science*, vol. 23, no. 6, pp. 785–806, 2004.
- [12] D. L. Gardner, L. S. Mark, J. A. Ward, and H. Edkins, "How do task characteristics affect the transitions between seated and standing reaches?" *Ecological Psychology*, vol. 13, no. 4, pp. 245–274, 2001.
- [13] O. Khatib, E. Demircan, V. D. Sapio, L. Sentis, T. Besier, and S. Delp, "Robotics-based synthesis of human motion," *Journal of Physiology-Paris*, vol. 103, pp. 211 – 219, 2009.
- [14] V. Sapio, J. Warren, and O. Khatib, "Predicting reaching postures using a kinematically constrained shoulder model," in *Advances in Robot Kinematics*, J. Lennarcic and B. Roth, Eds. Springer Netherlands, 2006, pp. 209–218.
- [15] A. K. Pandey and R. Alami, "Towards task understanding through multi-state visuo-spatial perspective taking for human-robot interaction," in *IJCAI Workshop on Agents Learning Interactively from Human Teachers (IJCAI-ALHT)*, July 2011.
- [16] A. K. Pandey, J.-P. Saut, D. Sidobre, and R. Alami, "Towards planning human-robot interactive manipulation tasks: Task dependent and human oriented autonomous selection of grasp and placement," in *IEEE RAS/EMBS BioRob*, 2012, pp. 1371–1376.
- [17] T. Simeon, J.-P. Laumond, and F. Lamiraux, "Move3d: a generic platform for path planning," in *4th Int. Symp. on Assembly and Task Planning*, 2001, pp. 25–30.