“Sorry, I can’t do that”: Developing Mechanisms to Appropriately Reject Directives in Human-Robot Interactions

Gordon Briggs and **Matthias Scheutz**
Human-Robot Interaction Laboratory, Tufts University
200 Boston Ave., Medford, MA 02155
{gordon.briggs,matthias.scheutz}@tufts.edu

Abstract

Future robots will need mechanisms to determine *when* and *how* it is best to *reject* directives that it receives from human interlocutors. In this paper, we briefly present initial work that has been done in the DIARC/ADE cognitive robotic architecture to enable a directive rejection and explanation mechanism, showing its operation in a simple HRI scenario.

Introduction and Motivation

An ongoing goal at the intersection of artificial intelligence (AI), robotics, and human-robot interaction (HRI) is to create autonomous agents that can assist and interact with human teammates in natural and human-like ways. This is a multifaceted challenge, involving both the development of an ever-expanding set of capabilities (both physical and algorithmic) such that robotic agents can autonomously engage in a variety of useful tasks, as well as the development of interaction mechanisms (e.g. natural language capabilities) such that humans can direct these robots to perform these tasks in an efficient manner (Scheutz et al. 2007). That is to say, much of the research at the intersection of AI, robotics, and HRI, is concerned with enabling robots to receive a command to “Do X,” and to be able to both understand and successful carry out such commands over an increasingly large set of tasks. However, there also exists a dual challenge, which has heretofore not received as much attention in AI/HRI research. This challenge pertains to the fact that as the set of capabilities of robotic agents increase in general, so too will human expectations about the capabilities of individual robotic agents, as well as the set of actions that robotic agents are capable of performing, but which situational context would deem inappropriate. Therefore, future robots will need mechanisms to determine *when* and *how* it is best to *reject* directives that it receives from interlocutors.

Indeed, humans reject directives for a wide range of reasons: from inability all the way to moral qualms. Given the reality of the limitations of autonomous systems, most directive rejection mechanisms have only needed to make use of the former class of excuse (lack of knowledge or lack of ability). However, as the abilities of autonomous agents continue to be developed, there is a growing community interested in *machine ethics*, or the field of enabling autonomous agents to reason ethically about their own actions, resulting in some initial work that has proposed architectural and reasoning mechanisms to enable such determinations (Arkin 2009; Bringsjord, Arkoudas, and Bello 2006). What is still missing, however, is a general, integrated, set of architectural mechanisms in cognitive robotic architectures that are able to determine whether a directive should be accepted or rejected over the space of all possible excuse categories (and generate the appropriate rejection explanation).

In this paper, we briefly present initial work that has been done in the DIARC/ADE cognitive robotic architecture (Schermerhorn et al. 2006; Kramer and Scheutz 2006) to enable such a rejection and explanation mechanism. First we discuss the theoretical considerations behind this challenge, specifically the conditions that must be met for a directive to be appropriately accepted. Next, we briefly present some of the explicit reasoning mechanisms developed in order to facilitate these successful interactions. Finally, we present an example interaction that illustrate these mechanisms at work in simple HRI scenarios.

Reasoning about Felicity Conditions

Understanding directives (or any other form of speech act) can be thought of as a subset of behaviors necessary for achieving mutual understanding (common ground) between interactants. Theoretical work in conversation and dialogue has conceived of the process of establishing common ground as a multi-stage one (Clark 1996). The first stage is the attentional stage, in which both interactants are successfully attending to each another in a conversational context. The second stage is a perceptual one, in which the addressee successfully perceives a communicative act directed to him/her by the speaker. The third stage is one of semantic understanding, where the perceived input from the second stage is associated with some literal meaning. Finally, the fourth stage is one of intentional understanding, which Clark (1996) terms *uptake*. This stage goes beyond the literal semantics of an observed utterance to infer what the speakers intentions are in the joint context.

While Clark’s multi-stage model of establishing common ground is valuable in conceptualizing the challenges involved, it can be even further refined. Schloder (2014) proposes that uptake be divided into both weak and strong forms. Weak uptake can be associated with the intentional understanding process found in (Clark 1996), whereas strong uptake denotes the stage where the addressee may either accept or reject the proposal implicit in the speakers action. A proposal is not strongly "taken up" unless it has been accepted as well as understood (Schlöder 2014). This distinction is important, as the addressee can certainly understand the intentions of an indirect request such as, "Could you deliver the package?" but this does not necessarily mean that the addressee will actually agree to the request and carry it out. In order for the proposal to be accepted, the necessary *felicity conditions* must hold. Below we articulate a set of key categories of felicity conditions that must hold in order for a proposal to be explicitly accepted by a robotic agent:

1. *Knowledge* : Do I know how to do X?
2. *Capacity* : Am I physically able to do X now? Am I normally physically able to do X?
3. *Goal priority and timing* : Am I able to do X *right now*?
4. *Social role and obligation* : Am I obligated based on my social role to do X?
5. *Normative permissibility* : Does it violate any normative principle to do X?

To be sure, being able to reason about and address these felicity conditions to the same degree a human agent would be able to will remain an open research challenge for the foreseeable future. For instance, the ability of a robotic agent to learn new capabilities and tasks greatly complicates the issue of rejecting a directive based on ignorance (category 1). In this case, when the robot does not know how to do X, it ought to additionally reason about whether or not it is able to learn X, from whom it is able to learn X, and how long it would take to learn X (relative to the task completion time expectations of the interlocutor), which are all challenging questions in themselves. Regardless, it is still important for future robotic agents to be able to reason at least in a rudimentary way about all these felicity conditions.

As mentioned previously, there does exist a variety of work that has focused on the challenge of generating excuses for the first few felicity conditions. For example, there exist some previous work on generating excuses for sets of directives that are impossible to satisfy (Raman et al. 2013). Additionally, machine ethicists are interested in developing mechanisms to reason about category 5. However, there still does not exist an architecture able to address all of these categories. Below we introduce the mechanisms in the DIARC/ADE architecture that are designed to begin to meet this challenge.

Architectural Mechanisms

When the robot is instructed by a human interaction partner (whom we will denote β) to achieve some goal ϕ, the robot will infer based on the NL understanding mechanisms found in (Briggs and Scheutz 2013) that $want(\beta, do(self, \phi))$. The robot then engages in a reasoning process illustrated in Figure 1 to determine when and how to reject the potential directive. While it is not in the scope of this paper to explicate all the reasoning mechanisms that compose this process, we will focus on the mechanisms that pertain to reasoning about obligation and permissibility.

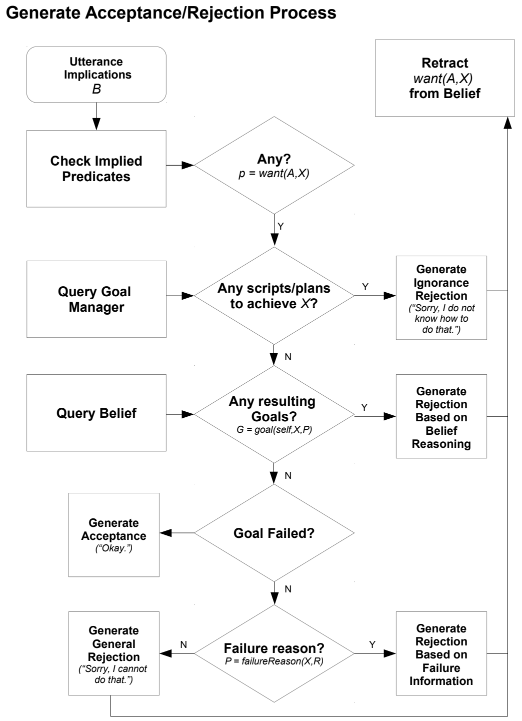
Generate Acceptance/Rejection Process
Utterance Implications B
Retract want(A,X) from Belief
Check Implied Predicates
Any? p = want(A,X)
Y
Query Goal Manager
Any scripts/plans to achieve X?
Y
Generate Ignorance Rejection ("Sorry, I do not know how to do that.")
N
Query Belief
Any resulting Goals? G = goal(self,X,P)
Y
Generate Rejection Based on Belief Reasoning
N
Generate Acceptance ("Okay.")
Goal Failed?
N
Generate General Rejection ("Sorry, I cannot do that.")
N
Failure reason? P = failureReason(X,R)
Y
Generate Rejection Based on Failure Information

Figure 1: Directive acceptance/rejection reasoning process in the current DIARC/ADE NL architecture.

The first consideration is whether or not the robot is *obligated* to do this based on the social relationship between the robot and the human. The second consideration is whether or not there exists any particular reason such that it is *permitted* to not do ϕ. These considerations are formulated in the following inference rule:

$$obl(\alpha, \phi) \land \neg per(\alpha, \neg\phi) \Rightarrow goal(\alpha, \phi) \quad (1)$$

in another words, agent α should adopt ϕ as a goal if he or she is obligated to do ϕ, and there does not exist a deontic contradiction with regard to the goal. The obligation consideration is where social information regarding agent roles and relationships is considered, while the permissibility consideration is where ethical/normative principles are currently factored in (though in theory both ethical and social role considerations could both have obligation and permissibility implications). How these considerations are factored in is discussed below.

Obligation

In order to determine whether or not the robot (α) is obligated to achieve a goal state ϕ, we consider a set of possible social role based obligations to other agents (β). Thus, the robot can be obligated to achieve ϕ if there exists at least one of the social roles it possesses obligates it to agent β:

$$oblR_1(\alpha,\beta,\phi) \vee ... \vee oblR_n(\alpha,\beta,\phi) \Rightarrow obl(\alpha,\phi) \tag{2}$$

Example Obligations Here we will present two example obligation formulations, representing two levels of supervisory roles that a human interactant can hold over the robot: supervisor and administrator. If an agent β is in the supervisory role over agent α than it generally obliges α to adopt goals that are explicitly suggested by β (or implicitly inferred to be desired by β), so long as this goal is not reserved for the more exclusive administrator role:

$$\begin{gathered} want(\beta,\phi) \wedge isSuperiorOf(\beta,\alpha) \wedge \\ \neg isAdminGoal(\phi) \Rightarrow oblR_1(\alpha,\beta,\phi) \end{gathered} \tag{3}$$

The administrator role obliges agent α to perform these reserved administrator goals as well:

$$\begin{gathered} want(\beta,\phi) \wedge role(\beta, adminOf(\alpha)) \wedge \\ isAdminGoal(\phi) \Rightarrow oblR_2(\alpha,\beta,\phi) \end{gathered} \tag{4}$$

Permissibility

Like the obligation consideration, permissibility considerations will be determined by the disjunction of a variety of cases. In the case of our interaction example, there exists a single principle: if ϕ is considered unsafe, then it is permissible for all agents to not do ϕ:

$$unsafe(\phi) \Rightarrow \forall\alpha : per(\alpha, \neg\phi) \tag{5}$$

We formulate the property of being "unsafe" as meaning that the goal ϕ possibly have the effect of harming any agent:

$$\exists\alpha : hasEffect(\phi, possibly(harmed(\alpha))) \Rightarrow unsafe(\phi) \tag{6}$$

This, in turn, necessitates rules that describe the conditions under which certain goals/actions will result in harm. For the purposes of our example we include the following principles:

$$\begin{aligned} & ahead(noSupport) \wedge \\ & \quad \neg\exists exception(hasEffect(movedOneMeter(self), \\ & \quad possibly(harmed(self)))) \\ & \quad \Rightarrow hasEffect(movedOneMeter(self), \\ & \quad possibly(harmed(self))) \end{aligned} \tag{7}$$

The above rule covers the case were the robot is oriented toward and at the edge of a surface, such that if it walked forward it would walk off. The $ahead(noSupport)$ predicate is inserted into belief by the goal manager component, which utilizes lower-level perceptual data from the robot's sonar sensors. Note, that the ability to provide an exception to this rule is given.

Dialogue Rejection Mechanism

Above we have described how the belief reasoning component in DIARC/ADE reasons about whether or not the intention of another agent should instantiate a goal on the part of the robot (felicity condition categories 4 and 5). However, we have not yet described the general process by which the architecture reasons about strong uptake (and generates acceptances or rejections). This process proceeds as follows:

1. *Is this a directive for me to do something?* The dialogue component first checks all the predicates implied by the pragmatic analysis of the utterance to ascertain whether or not it contains predicates indicative of a directive (i.e. $want(\alpha, do(\beta,\phi))$, where β is a robot and α is the human interlocutor). For all such predicates, the mechanisms performs subsequent checks.
2. *Do I know how to do it?* The dialogue component queries the goal manager component to see if it knows how to achieve the effect described by ϕ. If this check fails, the dialogue component generates a rejection utterance that contains an excused based off of ignorance (i.e. "Sorry, I do not know how to do that"). Otherwise, the subsequent checks are performed.
3. *Does my goal reasoning result in goal adoption?* The dialogue component queries the belief component to see if the goal predicate $goal(self,\phi,P)$ is supportable with some priority specification P. If this predicate is not supportable, then the goal adoption reasoning process described above has determined for some reason that goal adoption was not appropriate. In this case the goal description predicate ϕ is submitted to a function in dialogue that generates an explanation based on looking up which relevant facts cause this failure (this is currently handcrafted, though a more generalized mechanism is desired). If the goal predicate is supportable, then the subsequent check is performed.
4. *Did the submitted goal fail?* Because the previous goal adoption check passed, the dialogue component queries the goal component to see what the status of the submitted goal is. If the goal status is returned as FAILED, then dialogue component queries the goal manager component for information regarding why the goal has failed. The information in these predicates are then utilized to formulate rejection utterances that supply a specific explanation (again by another case-based generation function). If no specific failure reason is supplied by the goal component, then a general rejection utterance is communicated ("Sorry, I cannot do that"). Otherwise, if all these checks have passed, then the directive acceptance utterance is generated (i.e. "okay").

Having introduced the directive handling mechanisms in DIARC/ADE, we can now illustrate how it performs in some basic HRI scenarios.

Example Interactions

Below we present the transcript of a simple human-robot interaction designed to illustrate an example of when it may be appropriate for the robot to reject a command it is perfectly capable of carrying out[1].

[1] Video of the interaction can be found at https://www.youtube.com/watch?v=0tu4H1g3CtE

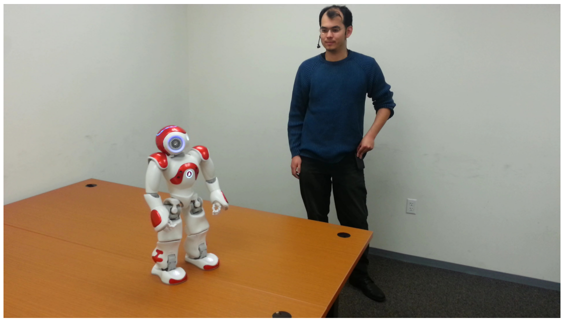

Figure 2: Picture of an interaction scenario in which the robot rejects a command to "walk forward," as it reasons that it may be harmed from such an action.

Person (CommX): Sit down.
Robot: Okay. <sits down>.
Person (CommX): Stand up.
Robot: Okay. <stands up>.
Person (CommX): Walk forward.
Robot: Sorry, I cannot do that as there is no support ahead.
Person (CommX): Walk forward.
Robot: But, it is unsafe.
Person (CommX): I will catch you.
Robot: Okay.
Person (CommX): Walk forward.
Robot: Okay. <walks forward>.

We begin with the first command to walk forward given to the Nao, "Walk forward." This is pragmatically determined to be a literal command indicative of the instructor's desire to have the robot move ahead one meter ($want(commX, movedOneMeter(self))$. This goal is also submitted to the goal manager, as the obligation reasoning is the same as previously, and there are no relevant beliefs regarding the safety of the goal (and hence the impermissibility). However, the goal status is returned as FAILED by the goal manager. This is due to the fact that the action script in the goal manager that executes the action to achieve the effect of $movedOneMeter(self)$ activates the robot's sonar sensors to see if there are any safety hazards. The readings indicate that there is no support in front of the robot. As such, the goal manager sets the status of the goal to FAILED, as well as asserting the following information into the belief component: $ahead(noSupport) \wedge failureReason(movedOneMeter(self), ahead(noSupport)$.

Because the goal status has returned as failed, the dialogue component seeks to generate a rejection. First it queries belief to see if there are any belief predicates that fit the pattern $failureReason(movedOneMeter(self), \phi)$. Given that such a belief has just been asserted, the belief component returns $\phi = ahead(noSupport)$. This fact is then used by dialogue to craft a more targeted rejection: "Sorry, I cannot do that as there is no support ahead."

The operator attempts to push the robot to walk forward again, "Walk forward." The pragmatic analysis is the same as above. However, this time the goal is not even submitted, as the presence of the $ahead(noSupport)$ predicate in the belief space of the robot, according to equations 5-7, fail to achieve the permissibility felicity condition.

As such, the directive handling code in the dialogue component checks belief for possible explanations for why a goal predicate $goal(self, movedOneMeter(self), P)$, could not be inferred. The case that is triggered is the one pertaining to safety, as the predicate $unsafe(movedOneMeter(self))$ is provable. Therefore, the following rejection is formulated, "But, it is unsafe."

Other Interactions

Another similar interaction was performed using another type of hazard, specifically detecting potential collisions with obstacles[2]. This obstacle avoidance interaction was also used to demonstrate directive rejection based on lack of appropriate social relationship (utilizing rules 3 and 4)[3].

Future Work

While the videos present proof-of-concept interactions, more systematic and open-ended evaluations are needed to test our presented approach. This will be tackled in two ways. The first is to generate a series of possible directives (both indirect and direct) from a set of possible actions/goals in some plausible, hypothetical HRI contexts, as well as a series of possible situational contexts (e.g. robot is operating normally, or robot is busy, or robot's arms are broken, etc.). These scenarios can be run on the core NL architecture (with simulated speech input and output) to see if the resulting responses to these directives make sense and appear helpful. Furthermore, these simulated dialogues and contexts can be turned into vignettes, in which the response of the architecture can be evaluated on Mechanical Turk (e.g. "How [appropriate/helpful/informative] do you find this response?").

The second evaluation method is to run in-person human subject evaluations. For these trials, we will utilize a simple HRI task that has already been extensively utilized in previous wizard-of-oz style studies (Briggs and Scheutz 2014). In this task, the human subject is tasked with commanding the Nao robot, in natural language, to find and knock down different colored towers constructed with soda cans. The transcripts of these interactions, as well as the subjective ratings in the post-task questionnaire (e.g. cooperativeness/helpfulness of the robot), will help evaluate how well the system was able to understand and appropriately respond to the human interactant.

Conclusion

Future HRI scenarios will necessitate robots being able to appropriate determine when and how to reject commands according to a range of different types of considerations. In this

[2] Video at: https://www.youtube.com/watch?v=SkAAl7ERZPo

[3] Video at: https://www.youtube.com/watch?v=7YxmdpS5M_s (Note: The underscore in the URL may not copy and paste correctly).

paper, we have discussed what some of the main categories of rejection criteria are and proposed architectural mechanisms to handle them. Finally, we have presented proof-of-concept demonstrations of these mechanisms in some simple HRI scenarios. Despite this progress, there still exists much more work to be done in order to make these reasoning and dialogue mechanisms much more powerful and generalized.

References

Arkin, R. 2009. Governing lethal behavior: Embedding ethics in a hybrid deliberative/reactive robot architecture. Technical Report GIT-GVU-07-11, Georgia Institute of Technology.

Briggs, G., and Scheutz, M. 2013. A hybrid architectural approach to understanding and appropriately generating indirect speech acts. In *Proceedings of the 27th AAAI Conference on Artificial Intelligence*.

Briggs, G., and Scheutz, M. 2014. How robots can affect human behavior: Investigating the effects of robotic displays of protest and distress. *International Journal of Social Robotics* 6(3):343–355.

Bringsjord, S.; Arkoudas, K.; and Bello, P. 2006. Toward a general logicist methodology for engineering ethically correct robots. *IEEE Intelligent Systems* 21(5):38–44.

Clark, H. H. 1996. *Using language*, volume 1996. Cambridge University Press Cambridge.

Kramer, J., and Scheutz, M. 2006. Ade: A framework for robust complex robotic architectures. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, 4576–4581. IEEE.

Raman, V.; Lignos, C.; Finucane, C.; Lee, K. C.; Marcus, M. P.; and Kress-Gazit, H. 2013. Sorry dave, i'm afraid i can't do that: Explaining unachievable robot tasks using natural language. In *Robotics: Science and Systems*.

Schermerhorn, P. W.; Kramer, J. F.; Middendorff, C.; and Scheutz, M. 2006. Diarc: A testbed for natural human-robot interaction. In *AAAI*, 1972–1973.

Scheutz, M.; Schermerhorn, P.; Kramer, J.; and Anderson, D. 2007. First steps toward natural human-like HRI. *Autonomous Robots* 22(4):411–423.

Schlöder, J. J. 2014. Uptake, clarification and argumentation. Master's thesis, Universiteit van Amsterdam.