Advances in Cognitive Systems 10 (2022) 1–20 Submitted 9/2022; published 11/2022

An Integrated Architecture for Common Ground in Collaboration

Christopher Geib CGEIB@SIFT.NET
Smart Information Flow Technologies

Denson George DENSON.GEORGE@RUTGERS.EDU
Baber Khalid BABER.KHALID@RUTGERS.EDU
Richard Magnotti RICHARD.MAGNOTTI@RUTGERS.EDU
Matthew Stone MATTHEW.STONE@RUTGERS.EDU
Department of Computer Science, Rutgers University, Piscataway NJ

Abstract

Effective teamwork depends on teammates' ability to maintain common ground: mutual knowledge about the relevant state of the world and the relevant status of teammates' actions and plans. This ability integrates diverse skills of reasoning and communication: agents can track common ground by recognizing and registering public updates to ongoing activity, but when this evidence is incomplete, agents may need to describe what they are doing or ask what others are doing. In this paper, we introduce an architecture for integrating these diverse skills to maintain common ground in human–AI teamwork. Our approach offers unique advantages of simplicity, modularity, and extensibility by leveraging generic tools for plan recognition, planning, natural language understanding and generation, and dialogue management. Worked examples illustrate how linguistic and practical reasoning complement each other in the realization of key interactive skills.

1. Introduction

Talk and collaboration go hand in hand (Clark, 1996). People's use of language is a collaborative activity: speakers and their interlocutors normally communicate as part of a joint effort to reach a shared understanding of how things are. Listeners, for example, regularly take the initiative to confirm, query, or reformulate speakers' references (Clark & Wilkes-Gibbs, 1986). At the same time, people's practical collaborations are normally supported by rich forms of communicative interaction. People use words, gestures, and other signals both to disambiguate their own actions and to ensure their understanding of others' contributions (Clark & Krych, 2004).

An important tradition of AI research has also pursued integrated accounts of communication and collaboration, beginning with the foundational work of Cohen & Perrault (1979) and Allen & Perrault (1980). Models of collaborative agency, for example, call for communication when observations are incomplete, results are unexpected, or goals change (Cohen & Levesque, 1991). Models of discourse, meanwhile, make explicit the joint goals that interlocutors pursue through talk and the relationships of those goals to their communicative acts (Grosz & Sidner, 1990; Lochbaum, 1998). Such work offers powerful explanatory principles but little guidance about how to start to

© 2022 Cognitive Systems Foundation. All rights reserved.

address the challenges of uncertainty, scalability, and system integration involved in putting those principles into practice.

Though often inspired by such integrative work, system-building efforts in AI have tended to focus either on systems' abilities to contribute to collaborative conversation or on their abilities to play a part in practical collaboration. For example, systems for practical collaboration offer scalable and robust methods to reason about teammates' actions and the system's own actions in the joint execution of complex activities (e.g. Kantharaju et al., 2019; Rich et al., 2001). However, these systems deal with ambiguity and problematic situations with scripted strategies rather than generative language use. Conversely, fielded dialogue systems may be able to disambiguate user requests using sophisticated policies that trade off the likelihood of errors against the awkwardness and risk of asking for repetition and clarification (e.g. Paek & Horvitz, 2000; Williams & Young, 2007). However, those systems generally cannot act overtly or in coordination with their users, and are unable to reason collaboratively about action. The limited capabilities of practical systems can often result in failure in simple scenarios where possible solutions seem intuitively obvious to people (see, e.g. Cohen, 2020; Larsson, 2017).

We believe that integrative reasoning about communication and practical collaboration is essential to achieving intelligent action in complex, open domains where systems must work effectively and collaboratively with people. A key aspect of this reasoning is maintaining collaborative common ground. Common ground constitutes a body of information that is publicly shared among the team and sufficient to resolve ambiguities and guide coordination in joint activity. Common ground has long been argued to play a fundamental role both in human–human interaction (Lewis, 1969; Clark & Marshall, 1981) and in AI teamwork (Cohen & Levesque, 1991), allowing team members to stay in sync as they assess the state of the world, decide what actions are necessary, and appraise their results.

In this paper, we focus specifically on teammates' need to make their *plans* common ground—a particularly revealing and challenging case. Teammates must coordinate their plans but cannot directly observe them, so teammates can maintain common plans only through a combination of rich practical understanding and skilled communicative interaction. Concretely, when agents coordinate open-ended problem solving, we cannot expect them to align their choices spontaneously or with simple signals. Language is essential to express the varied information that collaborators will need to work together. At the same time, common ground is too rich to communicate fully. Communication must function as a supplement to collaborative reasoning that tracks teammates' information and expectations (Macke et al., 2021; Rich et al., 2001).

We begin by reviewing previous research in Section 2. Researchers have so far had limited success in tracking common ground across both practical activity and communication (Chai et al., 2016; DeVault, 2008; Galescu et al., 2018; Macke et al., 2021; Rich et al., 2001). In part, we argue, this is because researchers have adopted representations and architectures that required specific, regimented connections between utterances and practical activity. New advances in probabilistic plan recognition and dialogue management, however, have developed representations and reasoning tools that enable more lightweight, flexible, and efficient approaches.

In Section 3, we offer case studies to explain how common ground reasoning requires communicative and practical reasoning to be deployed in flexible ways. As in Khalid et al. (2020), we argue

for modeling communicative action at the level of the discourse—through updates and other inferences that apply holistically to content presented across multiple turns by multiple participants—not just at the level of the utterance. Meanwhile, as in Geib (2015), we argue for plan reasoning capabalities that allow for the opportunistic adoption of new goals and the pursuit of multiple plans in parallel. The payoff of these new abstractions is that they naturally accommodate more diverse pathways and strategies for providing and confirming evidence of understanding and achieving common ground in collaboration.

To substantiate our arguments, in Section 4, we describe the design of a generic dialogue manager that handles common ground reasoning by assessing relevant ambiguities in a model of public information and pursuing domain-general strategies to agree on a resolution of those ambiguities. We complement the dialogue manager with a communication-enabled module for tracking practical teamwork, so that it is possible to query and select for collaborative plans that satisfy additional constraints (specifically, constraints derived from dialogue content).

Section 5 summarizes and demonstrates an initial implementation of a flexible, modular architecture for common ground reasoning in human–agent teamwork based on this design. Our implementation shows concretely how existing software for dialogue management, collaboration, planning, and plan recognition can be deployed together, and confirms that the system exhibits the flexible reasoning we have aimed for. The implementation remains a proof-of-concept, however, and substantial work remains to scale up knowledge resources and processing to cover a full application domain and assess performance in practice. Section 6 presents our approach to these limitations and future work.

2. Related Work

Human interlocutors are extremely adept at common ground reasoning. They recognize their shared experiences and their copresent environment as common ground (Clark & Marshall, 1981) and track the implications of a wide variety of manifest real-world events on common ground (Clark & Krych, 2004). They organize their participation in conversation to make sure that contributions become common ground, notably by providing evidence of understanding before moving forward (Clark & Schaefer, 1989). This evidence can take a wide variety of forms, including not only spoken utterances (Clark & Wilkes-Gibbs, 1986) but also practical actions and demonstrations (Clark & Krych, 2004).

In the experiments of Clark & Krych (2004), for example, one subject instructs a second in the assembly of a Lego model. To make sure the assembly plan is common ground, the follower might display the next piece or poise it in position while asking a telegraphic question about an ambiguity they perceive; in response, the leader might confirm, elaborate, or react directly to the follower's proposed understanding. Such capabilities in part reflect the broad psychological category of Theory of Mind, which describes the ability of humans or machines to draw inferences about others' psychological states, such as knowledge, belief, desire, emotions, and intentions. See Langley et al. (2022) for a recent review of these capabilities from the standpoint of AI research. In particular, the leader and follower's contributions are closely informed by their inferences about their partner's likely commitments for the upcoming assembly procedure. This specific problem is known as inten-

tion recognition (Mirsky et al., 2021). Note that AI approaches to Theory of Mind often address a broader range of cognitive and affective attitudes (including, for example, standing preferences and beliefs not related to ongoing action), but draw comparatively coarse inferences (for example, recognizing only overall goals or high-level activities rather than detailed plans). Note also that Theory of Mind inferences in this task are highly uncertain and potentially problematic; people's success in maintaining common ground in this task depends on linking Theory of Mind to interactive abilities to confirm and correct their understanding of one another.

2.1 Collaborative Grounding in Discourse

Abilities like those documented by Clark & Krych (2004) have motivated a number of general agent architectures for conversation and collaboration (Cohen, 2020; DeVault, 2008; Galescu et al., 2018; McShane et al., 2021; Rich et al., 2001). Our work is inspired by and builds on these efforts, but addresses a number of their drawbacks.

The COLLAGEN architecture of Rich et al. (2001) was the first collaboration architecture to put common ground reasoning and collaborative planning front and center in human–AI interaction. COLLAGEN uses plan recognition to track observed activity, cutting down on the need for communication. When ambiguity remains, COLLAGEN can ask for clarification about the user's goals, actions, and plans. COLLAGEN's common ground reasoning is limited, however, because the system assumes all practical and communicative actions contribute to a single known goal and because the system cannot plan utterances systematically to resolve ambiguity in context. Recent approaches such as Cohen (2020) and McShane et al. (2021) offer more powerful plan representations but still cannot assess potential ambiguities or track potentially problematic interpretations across discourse. DeVault (2008), meanwhile, extends COLLAGEN's approach with general language understanding and generation mechanisms; with appropriate resources and interaction design, this reasoning can be made symmetrical across roles in interaction (McMahan & Stone, 2013). However, it continues to suffer from following COLLAGEN's constrained approach to plan recognition. In Section 3, we see how these weaknesses are surprisingly limiting in supporting flexible collaboration.

By contrast, Cogent (Galescu et al., 2018) demonstrates that well-designed architectures can enable the creation of powerful collaborative assistants using generic modules for problem solving and communication, without rich knowledge resources such as plan libraries connecting language and action. However, Cogent does not apply plan recognition to user action or relate status updates reported by practical problem solvers to a general model of common ground and communication. These shortcomings make the system's interaction cumbersome and inflexible in settings that require sophisticated common ground reasoning.

By comparison to these integrative approaches, specialized models of grounding and teamwork offer much more flexible capabilities. In research on common ground in conversation, approaches based on information-state update (Larsson & Traum, 2000) describe the communicative goals and obligations behind grounding moves such as acknowledgments and clarification in dialogue; meeting these goals and obligations creates interactions that reliably achieve common ground (Matheson et al., 2000), even when grounding moves are realized through gesture (Nakano et al., 2003) or practical action (Hough & Schlangen, 2017). Approaches based on probabilistic reasoning (Paek & Horvitz, 2000; Williams & Young, 2007; Chai et al., 2016) add the ability to calibrate strate-

gies for pursuing common ground—across modalities of communication—by trading off the risk of misunderstanding against the effort of disambiguation. Recently Macke et al. (2021) extend such approaches to querying user intentions in *ad hoc* teamwork. The key limitation of most work in this tradition, however, is its narrow focus on the system's ability to disambiguate and confirm *user* contributions. This tradition does not address how the system can anticipate the user's information needs or give guidance about the information the system should use to respond to clarification questions and other problematic communication initiated by the user.

In our recent work, we have been exploring an alternative approach to combine models of dialogue with approaches to collaborative interaction (Khalid et al., 2020). The key insight is to apply models of collaboration at the level of the discourse as a whole, rather than at the level of the individual utterance, as typical of previous work across all the traditions we have reviewed. To interpret the discourse, we attach utterances into an evolving discourse structure which determines a knowledge graph specifying the contributions of different interlocutors. To reason collaboratively, we link this knowledge graph to entities and other information from the task context and assess the implications for ongoing activity. In previous work, we have instantiated this model with simple referential communication tasks and showed that it allows for flexible, effective, and human-like interaction strategies with feasible resources and reasoning. The present paper represents our first attempt to explore this approach in the context of practical collaboration.

2.2 Advances in Plan Recognition

Plan recognition, meanwhile, has become a dynamic area of research, with increasingly powerful approaches to recognize agents' high-level goals, ongoing activities, and the complex plans that tie them together; see Mirsky et al. (2021). Our work builds most closely on LEXrec, the plan recognizer of Geib's (2015) Engine for LEXIcalized Reasoning (ELEXIR) system. LEXrec builds explanations that represent the plans to be recognized by by parsing action sequences with a plan grammar. It further links the explanation structures to a model of the state of the world that changes successively as the agent's actions are executed. Unlike neural net or other sub-symbolic methods, ELEXIR recognizer performs probabilistic plan recognition using weighted model counting of explanations for a set of observed actions. Geib et al. (2016) discusses the use of ELEXIR to recognize plans and support improvised human–AI collaborations in the context of a robot trying to help human users.

Given an initial state of the world model, a plan grammar, and a sequence of observations, LEXrec first builds multiple explanations consistent with the plan grammar that capture not only the steps and causal structure of the plan being followed by the agent but also the causal linkages between the actions (which can be used to explain the role of individual actions in larger plans) as well as the causal links from the plan to the state of the world (to explain why the overarching plan was undertaken in the first place). Thus, different explanations may have very different (even contradictory) accounts of plans being followed and the reasons and roles for various actions. Each such explanation has an associated probability computed by LEXrec's parsing algorithm during its construction. We note that the explanation probability is conditioned on the state of the world model. This allows the system to conclude that a person talking on a cellphone in front of a burning

building is more likely to be calling the fire department than ordering a pizza. Thus while LEXrec can produce both explanations, it can also infer the first to be more likely.

We will leave most of the details of LEXrec's explanation building through parsing to the prior cited papers. However, its algorithm does have a property necessary for this work that few other plan recognition systems have. LEXrec does not require that all of the observed actions contribute to a single plan. It is designed to consider the possibility that an agent is executing multiple concurrent and potentially interleaved plans. To do this, unlike most natural language parsing algorithms, LEXrec does not require strict adjacency in the non-terminals of its grammars. For example, if a plan grammar has a production that states that to achieve goal G_1 requires that an action of type B follow an action of type A ($G_1 \rightarrow A\ B$), LEXrec recognizes instances where the order of the actions is preserved but there was another action of say type C is executed between them. Thus, if LEXrec also had a production $G_2 \rightarrow C\ D$ in its plan grammar, and observed the action sequence $[A, C, B, D]$ it is able to recognize this as the agent executing a plan for G_1 and a plan for G_2 at the same time. While this is counter-intuitive for parsing natural languages, it is critical for parsing multiple interleaved plans and most of the common ground examples we will talk about here depend on it. It is also worth noting that there are parsing algorithms that achieve their efficiency by assuming a single plan is present. LEXrec also cannot make any use of this kind of assumption.

On the basis of the computed explanations, ELEXIR can compute the conditional probability of any query that can be expressed as a Boolean test on a given explanation. ELEXIR simply sums the probability mass of those explanations for which the test is true and divides that by the probability mass of the set of all the explanations. This gives it much greater flexibility and power than most other plan recognition systems. For example, because explanations are linked to world models in addition to plan models we can ask how likely a given plan is given an uncertain state of the world (ie. how likely is it that you plan to mow the lawn if I see that you're going to the garden shed but there's a 75% chance that it's raining?). In addition, through its specialized parsing algorithm and significant structure sharing, ELEXIR's approach is able to scale to process thousands of observations in seconds and track thousands of competing and possibly conflicting hypotheses at the same time (Kantharaju et al., 2019). All of these properties make ELEXIR ideally suited to this task.

3. The Integrative Nature of Common Ground Reasoning

In this section, we use a series of case studies to highlight a basic challenge of grounding: enabling systems to resolve potential ambiguities in a natural way. Most importantly, the system must be able to interact with users when any party faces an ambiguity that could potentially derail the interaction: it must be able to elicit clarifications from the user and answer questions about its own contributions. At the same time, excessive clarification is tedious and distracting; the system should ask for clarification only if there's a genuine ambiguity that matters. This means we must avoid the proliferation of "computer ambiguities" (Ernie Davis's term) that only arise because the system has incomplete expectations about the collaboration or fails to factor those expectations into its inferences.

Our examples are drawn from an idealized search-and-rescue scenario where the system offers decision support or practical assistance to people exploring a damaged building, clearing obstacles, and treating and evacuating survivors. (The scenario is inspired by the DARPA ASIST program.)

3.1 Phenomena

Example 1: Grounding effects of physical actions in discourse context. Central to flexible and efficient common ground reasoning is the ability to relate ongoing action to constraints expressed in language. The inferences go in both directions. In tracking common ground, collaborators use what has been said as public evidence for the larger plans behind observed actions. Conversely, collaborators can interpret practical actions as providing evidence to confirm that utterance content is common ground. Here is a very simple example in the search-and-rescue domain:

> Player 1: "Hey someone transport the victim in Room 23?"
> Player 2: Picks up the victim.
> An assistant should take it as common ground that Player 2 is complying with the request and plans to continue by transporting the victim as instructed. Building on that common ground, an assistant could provide Player 2 with relevant supporting information, such as where the victim is supposed to go.

As simple as this case is, it is challenging to handle through inferences that robustly lead to concise, flexible, and confident communication. In a realistic collaboration, many tasks may be ongoing or possibly relevant, and plan recognition will deliver multiple interpretations of practical actions. Frameworks like COLLAGEN (Rich et al., 2001) that posit a single overarching goal effectively define this ambiguity away. At the same time, a strong presumption of collaboration is that agents are proceeding as planned, so collaborators can be confident in interpretations that are consistent with the content of prior discourse and established goals (when there are any). Without a model of these connections and their input on common ground, systems like Cogent (Galescu et al., 2018) will only register what's said explicitly (and will need clarification in cases such as these).

In our example, one interlocutor has expressed a preference to a teammate, but the same logic is needed when one agent announces their intentions for their own future activity or answers a clarification request from a teammate. In all cases, the explanations that fit what has been said can be taken to be common ground. This underscores the need to handle such connections at the level of the discourse, where they can be handled in a uniform way, rather than through speech acts or similar dynamics at the level of individual utterances, where diversity abounds.

In addition, the symmetry of the interaction means a focus on confirming the system's own understanding, as is common in dialogue research, is only half the story. Collaborators, including systems, must also look for positive evidence that their contributions are understood and accepted by their interlocutors. Complying with an instruction provides such evidence. In general, collaborators can use demonstrations not only to show that they are ready to proceed with the task, but also—when accompanied with appropriate verbal or nonverbal cues (Clark & Krych, 2004)—to show that they want further confirmation that they have understood correctly. To track common ground, we must recognize that practical actions can advance team goals while simultaneously providing the basis for contributions to coherent discourse.

Example 2: Using recognized plan context to disambiguate utterances. In practical collaboration, teammates understand utterances as making contributions to a shared planning process. Psycholinguistic studies of human–human collaboration have shown how easily people take task context into account in resolving linguistic ambiguities (Hanna & Tanenhaus, 2004; Brown-Schmidt et al., 2004) and how fundamental it is to human practices for giving and following instructions (Crangle, 1989). For example, there is usually no need to consider referentially consistent interpretations that wouldn't advance current goals. These are very likely to be computer ambiguities that violate "common sense".

> There are both a critical and non-critical victim in Room 23 but Player 1 would only need help with the critical one, which requires the coordinated effort of two teammates.
> Player 1: "Can you give me a hand? I need to triage the victim in Room 23."
> An assistant should be confident that the victim in question is the critical victim. Even if there's some other reason to suspect that Player 1 is not be aware of both victims, clarification should build on this understanding—for example "You mean the critical victim? You can handle the other one right?"

The challenge here is that such cases defy efforts to interpret language at the level of individual expressions or utterances. Here, understanding requires resolving the reference of "the victim." But if we consider this expression on its own, we don't see the planning context needed to recover what's intended. We either make an unreliable prediction or start a redundant interaction to address the apparent ambiguity. The planning context is only available at the level of the discourse: Player 1 has just made a request for assistance and is now offering an elaboration or explanation of the goal for which two agents are needed. Grounding in practical collaboration thus depends on linking language to action at the level of the extended discourse.

Example 3: Clarification of contributions. Of course, sometimes even after you take all the available information into account, there can still be important ambiguities. Maintaining common ground requires the ability to initiate and track clarification subdialogues to resolve such ambiguities. The ability to clarify linguistic ambiguities such as lexical and syntactic ambiguities, referential ambiguities, and speech act ambiguities is a well-established capability of the dialogue systems reviewed in Section 2. In teamwork, however, the same considerations apply to understanding agents' actions as well as their utterances. Concretely, imagine an agent observing a player that performs a set of actions that have multiple possible interpretations each of which require a different response. If there are no observable features of the context that will disambiguate, then to maintain common ground about the player's intent, the agent needs to ask a question about what the player is doing:

> Player 1 follows a hall toward a junction. It matters which way they plan to go next.
> An assistant should recognize that Player 1's destination is not common ground and needs to be, should produce a clarification such as "Are you heading to the east or west wing?" It should be able to use Player 1's answer to update the common ground. With common ground achieved, the system could provide timely interventions.

Any test of common ground reasoning should combine the abilities to track information across the collaboration, as shown in Examples 1 and 2, with the ability to detect and resolve ambiguities as in Example 3.

Summary. Examples 1–3 use the logic of resolving ambiguity to show the importance of plan recognition in maintaining common ground. To interact naturally and efficiently, Examples 1 and 2 show, systems need to integrate information to build coherent interpretations of ongoing talk and activity—possibly across extended discourse—reflecting both linguistic context and task context. Moreover, as Example 3 shows, systems need to be able to detect missing information in both language and action and pursue communicative strategies—again, possibly across extended interactions—to align their understanding with their teammates'.

3.2 Characterizing Common Ground Reasoning

Our key examples give us the wherewithal to better characterize the reasoning needed to track and pursue the common ground status of collaborative plans. In particular, they set up important requirements for generic plan recognition and language understanding components to interface in common ground reasoning. The use of generic tools streamlines the design, implementation, testing, and domain adaptation of the software architecture. However, these modules must provide the fundamental capabilities to deal with ambiguity: the system must be able to maintain multiple hypotheses for user contributions across its understanding processes, to be able to apply constraints from across discourse to resolve plan recognition, and to apply constraints from plan recognition to resolve ambiguities in extended discourse. In characterizing common ground reasoning for plans, our key contribution is to highlight the interactions of discourse-level inference and plan recognition that is needed to handle even simple cases.

Example 1: Grounding effect of physical actions introduced by natural language. Tracking common ground—even for such simple cases as complying with an instruction—involves applying discourse constraints in explaining observed actions. For grounding, then, we need a plan recognition module that exposes potential ambiguities for further inference by providing multiple explanations for observed events. Since grounding often requires a system to take calculated risks about whether to move forward with ongoing activity in the presence of uncertainty, these explanations should be associated with probabilities. Given such representations, we can take discourse information into account by selecting a subset of explanations that are compatible with discourse constraints. We refer to this operation as *filtering* throughout this paper.

Note that for filtering to work, plan recognition must produce explanations that are compatible with the ways that human teammates describe their plans in terms of overarching, high-level goals, like "transporting the victim" in Example 1. Recognized interpretations of actions must also connect initial low-level observations, like picking up the victim in this case, to these high-level goals. At the same time, people can telegraph their plans by committing to representative future actions that make their intentions clear. In Example 3, for example, a player might indicate their next steps by saying "I'm going to triage the victim in room 17", "I'm going to the east wing", or even "I'm turning right at the next junction." These utterances might all resolve to signal the same plan. Obviously, in complex environments, these constraints might be expressed across a coherent sequence of utterances, which is why it is important to filter based on the discourse, not just individual contributions to conversation.

Example 2: Using recognized plan context to disambiguate utterances. Example 2 shows, conversely, that reasoning about linguistic ambiguities is constrained by the task context. Systems that interpret language in light of ongoing activity typically recognize the real-world entities evoked by linguistic references by means of a dedicated process we will call *anchoring* (also known, somewhat confusingly, as semantic grounding). Generally, the anchoring process is implicitly constrained so that any action content should fit the possible plans associated with ongoing activity—as in our example, where "helping with the victim" is only compatible with an explanation of the overall plan where the victim in question is a critical victim. Anchoring must therefore restrict interpretations based on task constraints.

Of course, language continues to filter task ambiguities as well. We conclude that common ground reasoning needs an *integrated* process of anchoring *and* filtering to deliver interpretations that make sense of extended discourse and practical activity simultaneously.

Example 3: Clarification of contributions. As we have seen, finding integrated interpretations can still leave ambiguities open. In such cases, systems must clarify. Generating perspicuous clarification requests requires the system to be able to introspect about what the ambiguity is and how best to pinpoint and resolve it. Concretely, using an example from the search-and-rescue domain, this involves establishing that a particular linguistic expression—say, "are you turning __?"—locates a point of difference across the different interpretations and that alternative descriptions—say "east or west"—distinguish the alternatives from one another. To make this happen, anchoring and filtering processes based on public information must be accessible from the natural language generation pathway as well as the natural language understanding pathway. The candidate interpretations from anchoring and filtering (represented in a data structure such as a set of bindings of values to variables) can then guide successive choices in generation.

3.3 Common Ground Reasoning and Generic Tools

Common ground reasoning to resolve ambiguity in carrying out complex tasks thus involves coordinating planning, plan recognition, and natural language dialogue, and requires sufficient functionality from system components. In particular, our examples show that any implementation of collaborative common ground reasoning must have plan recognition and planning capabilities that:

1. share a common domain representation for plan recognition and planning,
2. support multiple, probabilistic explanations of team contributions,
3. can restrict explanations using external constraints for anchoring and filtering, and
4. expose ambiguities and appraisals for language understanding, dialogue management, and language generation in support of common ground.

In fact, ELEXIR (Kantharaju et al., 2019) as currently implemented meets these requirements, substantiating our goal of using generic plan reasoning in support of common ground.

Similarly, collaborative common ground reasoning requires natural language understanding and generation capabilities that:

1. share a common semantic representation across understanding and generation that tracks how information accumulates across speakers and utterances to resolve ambiguities,
2. link interpretation to the linguistic and nonlinguistic context, enabling references to be anchored through domain reasoning at the level of the discourse, and
3. assess possible interpretations in the course of understanding and generation and guide generation and dialogue management to resolve ambiguities.

Khalid et al. (2020) offer one approach to generic natural language understanding and generation modules in dialogue that meets these requrements. The key idea is to use representations of the logical form of discourse (LF) inspired by linguistic work on segmented discourse representation theory (Asher & Lascarides, 2003). LF makes explicit how contributions to discourse fit together by showing how discourse units assert propositions, raise questions, and express preferences and how those units fit together into a hierarchical structure governed by principles of discourse coherence. Meanwhile, LF makes explicit how meaning depends on the linguistic and nonlinguistic context by including variables (commonly called *discourse referents*) which must be anchored to particular values in context by matching discourse constraints against available real-world information.

Khalid et al. (2020) describe the rationale for using the LF for discourse to guide grounding in referential communication and the evaluation of an implemented system using the approach. They use standard NLP techniques, including machine learning methods, to build understanding and generation methods over an LF for discourse; they describe a dialogue manager that anchors LF to entities from a contextual situation and manages the resulting ambiguities to achieve common ground for dialogue.

The challenge and opportunity is thus to combine these tools to manage common ground in an integrated way as teammates pursue their collaboration through both linguistic and practical contributions. The next section explains our approach.

4. Design Principles for Integrated Common Ground Reasoning

We have developed a system architecture and proof-of-concept demonstration for implementing common ground reasoning. A visual schematic of our system architecture is presented in Figure 1. At the heart of the architecture is the public state of the interaction. This state consists of integrated interpretations that make sense of language and action in combination, derived by anchoring and filtering from both the logical form of discourse and the plan-based explanations of practical activity. Because they integrate the outcomes of prior actions, the rationale for ongoing actions, and agents' commitments and expectations going forward, these interpretations can capture all the state information needed to coordinate effectively. In particular, agents have achieved common ground when mutually available information leaves only one such interpretation, and all parties to the conversation have committed to this interpretation.

At a high level, our architecture combines two interaction pathways. The top pathway handles dialogue. It accepts communicative actions from users, processes them using natural language understanding to update the logical form of the discourse (LF), semantically resolves the logical form using contextual information (in the anchoring and filtering step), and derives a representation

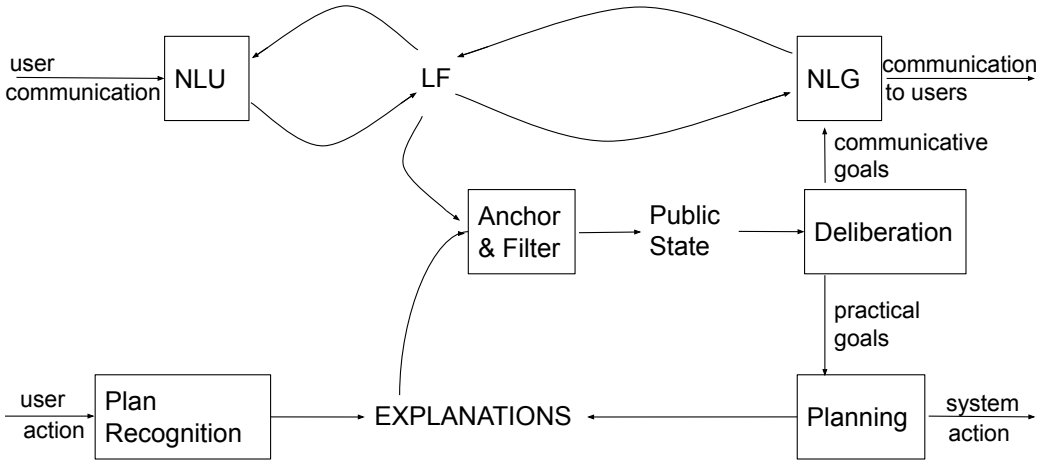

Figure 1. Our proposed system architecture for pursuing common ground in collaboration integrates dialogue and practical activity, correlates discourse information with plan-based explanations of observed actions, and treats system contributions and user contributions with symmetrical representations and reasoning.

of public information suitable for tracking the common ground. Communicative goals selected in deliberation return to this pathway via a natural language generation module that synthesizes utterances to achieve communicative goals, updates the logical form of the discourse and presents utterances to users. Our discussion of this language pathway explains the representations and reasoning used along this pathway and shows how the approach reproduces the capabilities of prior research on maintaining common ground in referential communication.

The bottom pathway, meanwhile, handles practical activity. It recognizes the possible plans behind user actions, and restricts those plans based on contextual information (in the anchoring and filtering step), so that the goal-directed nature of ongoing practical activity is also characterized in the system's representation of public information and common ground. Goals selected in deliberation return to this pathway via a planning module that can extend recognized plans to include system assistance. Our discussion of the action pathway describes the representations and reasoning used along this pathway and shows how the approach reproduces the capacities for collaborative agency and teamwork of previous systems built using ELEXIR.

Crucially, there are important points of contact and synergies across these two pathways. The process of contextual resolution applies holistically to resolving ambiguities in linguistic meaning and action explanations; this allows the architecture to support uniform strategies that the system can use to elicit clarifying information to maintain common ground in response to linguistic ambiguities or practical uncertainty. Similarly, deliberation can propose communicative goals and practical goals simultaneously, and planning and natural language generation can work in tandem to allow the system to describe its ongoing activity to teammates in concise but unambiguous ways.

4.1 The Language Pathway

The NLU module in Figure 1 is specified as taking as input the LF of the discourse so far and a new utterance, and as yielding a result LF that accounts for the full discourse by adding the content of the new utterance. This means that LF must be independent of the task domain and real-world context. For example, while LF specifies that discourse referents must be associated with real-world entities, it cannot resolve those referential connections itself. These are resolved in the anchoring process, which finds candidate instantiations for discourse referents given the constraints expressed in language and the contextual knowledge available from the situation and explanations.

In collaboration, a key role for LF is to track commitments to practical action. These commitments provide the basis for disambiguating practical ambiguity and coordinating joint action between teammates. Thus, it's important to understand how LF represents action content and what's involved in anchoring that action content to the context of practical activity.

Suppose in our search-and-rescue domain, one team member has decided to proceed to a particular room to triage critical victims there. Along the route, the team member reports "There's rubble here. I'm going around via the main hallway." The LF of this mini-discourse provides the intermediate representation that we will use to bridge between NLU and cooperative reasoning:

- "There's rubble here." LF classifies the coherent contribution this makes to the discourse: this is a summary recognizably distilling the import of the situation that the speaker is confronted with. It draws on the prominent place "here" established in that situation and introduces an entity "rubble" that (since this is a summary) should be recognizable by any viewer assessing the situation. So much is clear from the grammar. Anchoring this LF involves finding the situation the speaker is talking about (their environment in pursuing their plan), recognizing the rubble (drawing on relevant situation awareness, if available), and recognizing that this represents a problem or obstacle to the current plan (appraising the summary relation).

- "I'm going around via the main hallway." LF classifies this as a coherent description of a consequence of the rubble. It uses general background about the environment to identify the key feature of an alternative route (implicitly preserving the destination) and commits the speaker to following the route so described. Again, this follows from grammar. Anchoring this LF involves recognizing that the speaker has abandoned the previous plan and is now committed to a new plan and linking the speaker's description to specific real-world locations that will constrain the new route. This sets the stage to interpret ongoing and subsequent action by the speaker with reference to this commitment.

Reviewing the anchoring operations, the system's information may demand grounding linguistic content. If the system doesn't know the speaker's exact location and expects other teammates to navigate similar paths, it may want to clarify where the rubble is. If the system can't recognize which hallway is the main hallway, the system may want to propose and check an alternative description. However, as researchers on grounding have often emphasized, ambiguity itself is not a problem: if nobody is likely to pass nearby again, it's OK to live with some uncertainty about where the rubble is; and the system will find out where the main hallway is as soon as the speaker proceeds that way. Managing ambiguity thus depends not only on the uncertainties of linguistic interpretation but on the dynamics and import of real-world collaboration.

4.2 The Action Pathway

When the system observes user actions, it first must update its explanations of the ongoing activity. ELEXIR starts from a plan lexicon linking actions to categories; each category specifies a top-level goal along with other categories that accompany the action (either preceding it or following it). To perform plan recognition, ELEXIR parses the observed actions into plan structures meeting the requirements defined in the plan lexicon. It also establishes a probability distribution over these explanations to reason about the most likely goals and plans. The plan lexicon specifies only the basic goals for each action, so ELEXIR does not reason in an open-ended way about possible ways the collaboration could unfold.

All the potential explanations from ELIXIR are propagated to the anchoring and filtering step, where the explanations compatible with the commitments of the discourse are selected. The public state is then available for processes of deliberation. The selection of communicative goals is guided by the presence of ambiguity, as well as by further conditions that may require acknowledgement or follow up for effective teamwork (Cohen & Levesque, 1991), such as 1) whether the most recent action introduces a new plan that needs to be acknowledged, 2) whether any of the user's current plans conflict, and 3) whether there is a completion of the plans the system thinks it has recognized that will be successful. ELEXIR's planning capabilities, meanwhile, can be used as in Geib et al. (2016) to infer better ways for the team to achieve their goals—including new plans that involve action on the part of the system.

4.3 Synergistic Reasoning

Although we have motivated and explained our architecture in terms of two intuitive pathways, it is crucial that the architecture actually handles grounding via integrated representations and uniform mechanisms. These synergies are crucial to its potential generality and flexibility. In this section, we explain this further, anticipating the demonstration results we present below.

Consider asking a clarification question about the plan behind a user's action. Concretely, as in Example 3, we might need to clarify which of two paths the user intends to explore.

Here's how that's handled. We observe the user's initial action. ELEXIR comes up with two plans in context, each of which projects additional future movement and task actions. If the user is acting on their own initiative, there is no discourse context to constrain these plans, so both are reported as possible interpretations in the public state. This creates an opportunity for grounding, because the user's plan is not shared information. Assuming deliberation calculates that this ambiguity would lead to potential coordination failure (this could be a general policy or could reflect a domain-specific probabilistic calculation trading off the risks of allowing the uncertainty to persist versus the costs of resolving it through dialogue, as in Macke et al. (2021)), this creates a communicative goal of requesting information to clarify the user's two plans. Natural language generation takes this communicative goal as input and uses the anchoring and filtering computation to recognize that "are you going to turn east or west" elicits the needed information in an unambiguous way. The system asks the question and updates the logical form of the discourse. Now the user's response comes in: "I'm going east." Natural language understanding updates the logical form of the discourse, making explicit the constraint that the user has placed on subsequent activity (and

indicating that the utterance is a coherent answer to the system's question). This constraint is now presented to the anchoring and filtering step, and as a result the westward plan is eliminated from consideration. There is only one possible plan, ambiguity has been resolved, and the system and the user can be presumed to have reached common ground.

In fact, however, there are symmetries in the system between natural language generation and natural language understanding, and between plan recognition and planning. These symmetries mean that the basic architecture and reasoning apply to a number of apparently quite different cases in a fundamentally similar way.

To start, the same common ground reasoning mechanisms apply even if there's no clarification question. That is, once the user says "I'm going east," the ambiguity in their action is resolved and the information can be taken as grounded. In fact, the ambiguity is resolved no matter how the information is contributed to the discourse. If the user starts moving in response to an instruction from another teammate, "Go east", the discourse information will still be applied to anchor and filter the discourse and the explanations of ongoing activity, and the system will track that both the instruction and the user's response have been grounded.

Moreover, the system can use the same logic to ground its own activity. For example, suppose the system is controlling an assistive robot, and has decided to move the robot down the same ambiguous hall. This action feeds back into plan recognition, with the result that the system is aware that there are two explanations of its own action given public information: one where it intends to go east (its actual, private plan) and another where it intends to go west (which it will not do, but which an observer might nevertheless consider). In the absence of further discourse information—for example an earlier instruction filtering the possibilities—this ambiguity propagates to the public state, and again creates a potentially problematic ambiguity. Deliberation can trigger a communicative goal to resolve the ambiguity, in this case by relying on the system's private information (something the system must always be able to do for natural language generation, of course). That goal feeds into natural language generation where it might lead to the utterance "I'm going east." Updating the logical form of the discourse again triggers filtering and the prediction that the utterance suffices to ground the system's plan. These natural generalizations show the power of the representation, reasoning, and architecture we have developed.

5. Worked Examples

We have a proof-of-concept implementation that takes the form of a generic shell for an autonomous collaborative agent capable of linguistic and practical action. The shell is instrumented to track public information about ongoing activity through stages of perception and action. Its deliberative mechanisms are designed to allow designers to prioritize goals of maintaining common ground and to avoid ambiguity in system decisions. Instantiating the shell with specifications of activity in a particular domain, strategies for supporting and assisting users in domain activity, and models of domain-specific language use allows the shell to be customized to a target application. The result of this specialization is a cooperative assistant combining domain-specific cooperative expertise with domain-general support for tracking common ground. In particular, the shell is designed to ac-

commodate off-the-shelf components for natural language understanding (NLU), natural language generation (NLG), plan recognition, planning, and other key inference problems.

As a preliminary demonstration of the approach, we have formalized several scenarios in the search-and-rescue domain of Examples 1–3. The demo takes place in an environment where multiple hallways connect three different rooms. At the very start, the player (user) is in a starting room and has the option to equip a medical kit, a hammer or both. There are two other rooms: a room in the right hallway containing a victim and a room in the left halfway containing some rubble which needs to be cleared.

Scenario 1. The player is in the starting room at the start and equips both the medical kit and hammer. Then the system observes that the player leaves the room which causes the system to come up with two explanations: the user can either go and help the victim or clear the rubble. Due to this, the system asks a clarification question. The player answers that they are going to help the victim. This answer is incorporated into the LF of the discourse, and the possible explanations are updated by taking the new constraint into account in the anchoring and filtering step. The only remaining explanation for the player action is that they are going to help the victim as specified in their clarification answer, and common ground is achieved. After this, the player passes through different hallways to reach their destination and triages the patient. the system tracks their progress without further ambiguity or clarification.

Scenario 2. As before, the player is in the starting room at the start and equips both the medical kit, then leaves the room. This time, the player provides the verbal explanation that they plan to help the victim before the system asks a clarification. Here, too, this information is incorporated by the state tracking mechanism of discourse manager, and the possible explanations are updated by filtering with the player's commitments. Again, the system recognizes that the player's plan is common ground, and tracks the player's continued actions without ambiguity.

Scenario 3. This scenario considers the clarification of system actions. We have the same scenario as before, except that the system acts alongside the human player. The scenario now has a successful outcome only if the team is able to both triage the victim and clear the rubble.

The player is in the starting room but only equips the hammer this time. The system, meanwhile, equips both the hammer and the medical kit. When the player leaves the starting room the only explanation can be that they are going to clear the rubble. The system therefore plans to triage the victim, and leaves the room. The system calculates, however, that public information leaves open two explanations for its action; its plans are no longer common ground.

To rectify this, the system provides the player with the information that it is going to triage the patient. This answer is incorporated by the state tracking mechanism of discourse and the only possible explanation which remains is that the player is going to clear the rubble and the system is going to triage the patient. Common ground is restored. No further ambiguity arises and no communication is needed as both the player and system navigate the hallways to reach their respective destinations and achieve their respective goals.

6. Discussion, Limitations, and Future Work

Substantial effort remains to expand our proof-of-concept into a broadly useful platform for building interactive systems comparable to COLLAGEN (Rich et al., 2001) or Cogent (Galescu et al., 2018). For example, our infrastructure for dialogue is based on simple referential communication tasks (Khalid et al., 2020) and needs more robust and flexible modules for natural language understanding, generation, and dialogue management to handle discussion of actions and plans. We need software development tools as well as run-time infrastructure, given the challenge of acquiring, managing, and profiling knowledge assets such as plan libraries, syntactic and semantic resources for language, and dialogue management policies. To support iterative refinement of common ground assistants, we also need tools to log, annotate, and analyze data from interactions with the system.

We are alert to a number of possible pain points that this development is likely to encounter. Our approaches to plan representation and reasoning may not be flexible enough to link inferred explanations to speakers' intuitive descriptions. More robust dialogue management and plan inference may be necessary to enable the system to recognize and recover from miscommunication and other errors. Finally, it remains to be seen what ambiguities arise in language and action in practical cases and whether collaborative reasoning and targeted interaction suffices to resolve them. Clearly, such questions can be answered only through extensive empirical evaluation.

We believe that the most compelling evaluation campaign will involve the creation of a virtual assistant system based on common ground reasoning in a novel application domain, created in collaboration with stakeholders who are considering the adoption of the technology and who bring meaningful tasks, clear definitions of success, and an engaged user population. Our plans for evaluation involve analyzing the experiences of developers and users to address three broad hypotheses that we believe must be confirmed to substantiate the system design: first, that common ground reasoning, as we have characterized it, contributes to team success; second, that our reasoning approaches succeed in maintaining common ground using creative, flexible, context-sensitive strategies; and third, that our software infrastructure offers efficient tools for building and computing common ground in new task and dialogue domains.

7. Conclusion

We have presented an architecture where language and action understanding are instrumented and wired together to deliver integrated interpretations of ongoing activity; these interpretations offer general resources to recognize common ground, as well as to detect, diagnose, and respond to problematic situations. We show how deliberative processes can draw on this input to create goals related to common ground maintenance alongside practical goals, and how planning processes can take interpretation into account so that the system reduces and avoids ambiguity in selecting actions to meet these goals. Our approach accomplishes its cycle of collaborative reasoning in a particularly general and flexible ways by using the analogous knowledge, representation, and inference for both the system's contributions and those of its collaborators.

Our work centers opportunistic and improvised collaborations. We work with lightweight plans as resources for coordination that underpin decisions made through diverse mechanisms. Such designs are very different from those of widely deployed personal digital assistants, which are inca-

pable of representing or reasoning about user knowledge, ignorance, and misconceptions, tracking extended activity involving multiple subgoals, or following flexible communication and problem solving (Cohen, 2020). But they also diverge from theoretical approaches (e.g. Cohen & Levesque, 1991; Grosz & Sidner, 1990), which postulate heavyweight plans that formalize all system decisions. The diversity of approaches to collaboration and the importance of the problem is sure to keep it an enduring challenge.

Acknowledgements

The authors would like to thank Jeff Rye for his assistance in using the ELEXIR codebase and its associated interfaces, and the reviewers for helpful feedback. This research was supported in part by NSF awards CCF-1934924, DGE-2021628, and IIS-2119265, and by Air Force Research Lab Contract FA8650-22-P-6415.

References

Allen, J. F., & Perrault, C. R. (1980). Analyzing intention in utterances. *Artificial Intelligence*, *15*, 143–178.

Asher, N., & Lascarides, A. (2003). *Logics of conversation*. Cambridge.

Brown-Schmidt, S., Campana, E., & Tanenhaus, M. K. (2004). Reference resolution in the wild: Circumscription of referential domains by naive participants during an interactive problem solving task. In J. C. Trueswell & M. K. Tanenhaus (Eds.), *World-situated language use: Psychological, linguistic and computational perspectives on bridging product and action traditions*. MIT.

Chai, J. Y., Fang, R., Liu, C., & She, L. (2016). Collaborative language grounding toward situated human-robot dialogue. *AI Magazine*, *37*, 32–45.

Clark, H. H. (1996). *Using language*. Cambridge.

Clark, H. H., & Krych, M. A. (2004). Speaking while monitoring addressees for understanding. *Journal of Memory and Language*, *50*, 62–81.

Clark, H. H., & Marshall, C. R. (1981). Definite reference and mutual knowledge. In A. K. Joshi, B. L. Webber, & I. Sag (Eds.), *Elements of discourse understanding*, 10–63. Cambridge.

Clark, H. H., & Schaefer, E. F. (1989). Contributing to discourse. *Cognitive Science*, *13*, 259–294.

Clark, H. H., & Wilkes-Gibbs, D. (1986). Referring as a collaborative process. *Cognition*, *22*, 1–39.

Cohen, P. R. (2020). Back to the future for dialogue research. *The Thirty-Fourth AAAI Conference on Artificial Intelligence* (pp. 13514–13519). AAAI Press.

Cohen, P. R., & Levesque, H. J. (1991). Teamwork. *Nous*, *24*, 487–512.

Cohen, P. R., & Perrault, C. R. (1979). Elements of a plan-based theory of speech acts. *Cognitive Science*, *3*, 177–212.

Crangle, C. (1989). On saying 'stop' to a robot. *Language and Communication*, *9*, 23–33.

DeVault, D. (2008). *Contribution tracking: Participating in task-oriented dialogue under uncertainty*. Doctoral dissertation, Department of Computer Science, Rutgers University.

Galescu, L., Teng, C. M., Allen, J., & Perera, I. (2018). Cogent: A generic dialogue system shell based on a collaborative problem solving model. *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue* (pp. 400–409).

Geib, C. (2015). Lexicalized reasoning. *Proceedings of the Third Annual Conference on Advances in Cognitive Systems* (p. 19).

Geib, C. W., Weerasinghe, J., Matskevich, S., Kantharaju, P., Craenen, B. G. W., & Petrick, R. P. A. (2016). Building helpful virtual agents using plan recognition and planning. *Proceedings of the Twelfth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* (pp. 162–168).

Grosz, B. J., & Sidner, C. L. (1990). Plans for discourse. In P. Cohen, J. Morgan, & M. Pollack (Eds.), *Intentions in communication*, 417–444. Cambridge MA: MIT Press.

Hanna, J., & Tanenhaus, M. K. (2004). The use of perspective during referential interpretation. In J. C. Trueswell & M. K. Tanenhaus (Eds.), *World-situated language use: Psychological, linguistic and computational perspectives on bridging product and action traditions*. MIT.

Hough, J., & Schlangen, D. (2017). It's not what you do, it's how you do it: Grounding uncertainty for a simple robot. *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction* (pp. 274–282).

Kantharaju, P., Ontañón, S., & Geib, C. W. (2019). Scaling up CCG-based plan recognition via Monte-Carlo tree search. *IEEE Conference on Games* (pp. 1–8).

Khalid, B., Alikhani, M., Fellner, M., McMahan, B., & Stone, M. (2020). Discourse coherence, reference grounding and goal oriented dialogue. *Proceedings of the 24th Workshop on the Semantics and Pragmatics of Dialogue*.

Langley, C., Cirstea, B. I., Cuzzolin, F., & Sahakian, B. J. (2022). Theory of mind and preference learning at the interface of cognitive science, neuroscience, and AI: A review. *Frontiers in Artificial Intelligence*, *5*. From `https://www.frontiersin.org/articles/10.3389/frai.2022.778852`.

Larsson, S. (2017). User-initiated sub-dialogues in state-of-the-art dialogue systems. *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue* (pp. 17–22).

Larsson, S., & Traum, D. (2000). Information state and dialogue management in the TRINDI dialogue move engine toolkit. *Natural Language Engineering*, *6*, 323–340.

Lewis, D. K. (1969). *Convention: A philosophical study*. Harvard.

Lochbaum, K. E. (1998). A collaborative planning model of intentional structure. *Computational Linguistics*, *24*, 525–572.

Macke, W., Mirsky, R., & Stone, P. (2021). Expected value of communication for planning in ad hoc teamwork. *Thirty-Fifth AAAI Conference on Artificial Intelligence* (pp. 11290–11298).

Matheson, C., Poesio, M., & Traum, D. (2000). Modelling grounding and discourse obligations using update rules. *1st Meeting of the North American Chapter of the Association for Computational Linguistics*.

McMahan, B., & Stone, M. (2013). Training an integrated sentence planner on user dialogue. *Proceedings of the SIGDIAL 2013 Conference* (pp. 31–40).

McShane, M., English, J., & Nirenburg, S. (2021). Knowledge engineering in the long game of artificial intelligence: The case of speech acts. *Proceedings of the Ninth Annual Conference on Advances in Cognitive Systems*.

Mirsky, R., Keren, S., & Geib, C. W. (2021). *Introduction to symbolic plan and goal recognition*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers.

Nakano, Y. I., Reinstein, G., Stocky, T., & Cassell, J. (2003). Towards a model of face-to-face grounding. *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics* (pp. 553–561).

Paek, T., & Horvitz, E. (2000). Conversation as action under uncertainty. *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence (UAI)* (pp. 455–464).

Rich, C., Sidner, C. L., & Lesh, N. (2001). COLLAGEN: applying collaborative discourse theory to human-computer interaction. *AI Magazine*, *22*, 15–25.

Williams, J. D., & Young, S. (2007). Partially observable Markov decision processes for spoken dialog systems. *Computer Speech & Language*, *21*, 393–422.