2014 IEEE International Conference on Robotics & Automation (ICRA)
Hong Kong Convention and Exhibition Center
May 31 - June 7, 2014. Hong Kong, China

Interpreting Instruction Sequences in Spatial Language Discourse with Pragmatics towards Natural Human-Robot Interaction*

Juan Fasola and Maja J Matarić, *Fellow, IEEE*

Abstract— **We present a methodology for enabling service robots to interpret spatial language instruction sequences expressed through natural language discourse from non-expert users. As part of our approach, we propose a novel probabilistic algorithm for the automatic extraction of contextually and semantically valid instruction sequences from unconstrained spatial language discourse. Additionally, we present the design and implementation details of a procedure for reference resolution of anaphoric expressions encountered within the user discourse. Towards application of our human-robot interaction (HRI) methodology on robot platforms in practice with end users, we discuss a generalized procedure for transfer to physical systems and provide solutions for key pragmatic considerations including the generation of safe robot execution paths for both the robot and people in the environment. The paper concludes with an evaluation of our spatial language-based HRI framework implemented on a PR2 robot to demonstrate the generalizability and usefulness of our approach in real world applications.**

I. INTRODUCTION

For autonomous service robots to provide effective assistance in real-world environments, they will need to be capable of interacting with and learning from non-expert users in a manner that is both natural and practical for the users. In particular, these robots will need to be capable of understanding natural language instructions for the purposes of user task instruction, teaching, modification, and feedback. This capability is especially important in assistive domains, where robots are interacting with people with disabilities, as the users may not be able to teach new tasks and/or provide feedback to the robot by demonstration.

Spatial language plays an important role in instruction-based natural language communication [1, 13]. For example, a user might teach a household service robot the complex task "Put away the groceries", through natural language and by specifying the subgoals of the task individually, each represented by its own spatial language instruction (e.g., "Put the spices in the top cupboard on the left hand side of the kitchen", "Stow away all of the trash bags under the sink", "Place the vegetables in the bottom shelf of the refrigerator", etc.). As the user provides a series of spatially-oriented instructions to the robot, the user is engaging the robot in discourse. In such cases, the user may at any point refer to a previously introduced entity or object (e.g., household items, people, regions of space, etc.) and the robot must be capable of keeping track of the current discourse history in order to resolve such referential expressions. In addition, users may not adhere to the language model of the robot (the grammar) and therefore the robot must be capable of extracting and/or inferring the desired tasks expressed by the user for the robot to perform from the unconstrained natural language input.

Therefore, the ability for robots to parse and understand unconstrained spatial language in spoken communication, and to maintain an active discourse model, is critical for the interpretation of user-guided instructions to be successful.

In this paper, we extend upon our previous work [7, 8, 9] and present a methodology for enabling service robots to interpret spatial language instruction sequences expressed through natural language discourse from non-expert users. We propose a novel probabilistic algorithm for the automatic extraction of contextually and semantically valid instruction sequences from unconstrained spatial language discourse. Additionally, we present the design and implementation details of a procedure for reference resolution of anaphoric expressions encountered within the user discourse. Towards application of our human-robot interaction (HRI) methodology on robot platforms in practice with end users, we discuss a generalized procedure for transfer to physical systems and provide solutions for key pragmatic considerations including the generation of safe robot execution paths for both the robot and people in the environment. The paper concludes with an evaluation of our spatial language-based HRI framework implemented on a PR2 robot to demonstrate the generalizability and usefulness of our approach in real world applications.

II. RELATED WORK

The use and representation of spatial prepositions, and spatial language in general, in human-agent interaction scenarios has been investigated by previous work. Skubic et al. [6] developed a mobile robot capable of understanding and relaying static spatial relations in natural language instruction and production tasks. The use of computational field models of static relations has also been explored in the context of human-robot cooperation tasks [5], and for visually situated dialogue systems [12]. Our approach extends upon this related work by modeling not only static spatial relations, but also dynamic spatial relations (DSRs).

Recent work has, however, explored the use of dynamic spatial relations in the context of natural language robot instruction. Tellex et al. [3] constructed a probabilistic graphical model to infer spatial task/actions commanded through natural language for execution by a forklift robot.

* Research supported by National Science Foundation grants IIS-0713697, CNS-0709296, and IIS-1117279, and by the Office of Naval Research MURI grant N00014-09-1-1031.

J. Fasola is with the University of Southern California, Los Angeles, CA 90089 USA (e-mail: fasola@usc.edu).

M. J. Matarić is with the University of Southern California, Los Angeles, CA 90089 USA (e-mail: mataric@usc.edu).

978-1-4799-3684-7/14/$31.00 ©2014 IEEE

Similarly, Kollar et al. [4] presented a probabilistic approach for interpreting route directions using learned models of dynamic spatial relations. These approaches typically require the system designer to provide an extensive corpus of labeled natural language input for each new application context, without taking advantage of the domain-independent nature of spatial prepositions. In contrast, our approach develops novel, pre-defined templates for spatial relations, static and dynamic, that facilitate use and understanding across domains, and whose computational representations enable guided robot execution planning.

Methods for mapping natural language instructions onto a formal robot control language have also been developed by researchers using a variety of types of parsers, including those that were manually constructed [10, 11], learned from training data [15], and learned iteratively through interaction [16]. Among these examples, the work of Rybski et al. [10] and Matuszek et al. [15] relied on pre-defined robot behaviors as primitives, as opposed to spatial relations, which limits, if not restricts, the user's ability to introduce feedback modifications and/or constraints on robot execution of a specific primitive behavior. The work of Kress-Gazit et al. [11] and Cantrell et al. [16] mapped words to meanings based on dictionary-based rules. Our methodology employs domain-generalizable spatial relations as primitives, and probabilistic reasoning for the grounding and semantic interpretation of phrases, thereby enabling context-based instruction understanding.

III. Methodology for Spatial Language Discourse Interpretation

In this section we present our methodology for autonomous service robots to interpret spatially-oriented instruction sequences expressed through natural language discourse from non-expert users. We begin with a brief overview of our approach and software modules towards understanding spatial language instructions in human-robot interaction, detailed in previous work [7, 8, 9], and follow with the presentation of novel methods for the probabilistic extraction of contextually valid instruction sequences from natural language discourse, and for resolving anaphoric references expressed within the user discourse.

A. Spatial Language-Based HRI Framework

We have developed a framework for human-robot interaction that enables the interpretation of spatial language directive instructions, with and without constraints, by encoding spatial relations within the robot *a priori* as primitives. Our approach utilizes the semantic field model of spatial prepositions, proposed by O'Keefe [2], and introduces an extension to the model that enables the representation of dynamic spatial relations (DSRs) involving paths. The semantic field model is beneficial as it provides a continuous representation for determining the applicability of a given spatial relation (applied to specified figure and reference objects) at each point in the environment, which can be used for both probabilistic reasoning and task planning. Fig. 1 shows an example semantic field computed for the spatial relation *at* (utilizing a Gaussian function for proximity), used during robot task planning for the instruction "Go to my desk".

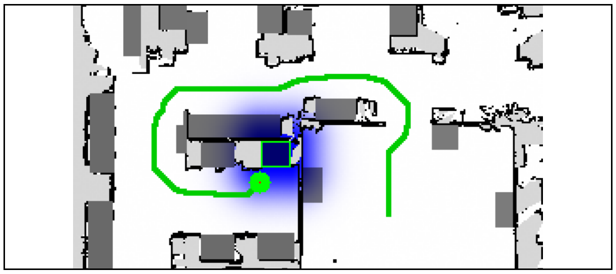


Figure 1. Semantic field for the spatial relation *at* applied to a desk reference object for the instruction "Go to my desk"

Our robot software framework is comprised of five system modules that enable the interpretation of natural language instructions, from user speech or text-based input, and translation into robot action execution. They are: the syntactic parser, noun phrase (NP) grounding, semantic interpretation, planning, and action modules. As discussed in the following subsections, the first three modules do not operate independently, but instead are integrated in a feedback loop designed to find the optimal interpretation for the natural language input given the context of the environment and the current discourse.

The semantic interpretation module accepts four observations as input: the verb and preposition utilized, and the associated grounding types for the expressed figure and reference objects (as returned by the NP grounding module). The module then infers the command semantics of the spatial instruction probabilistically using a Naïve Bayes approach over a database of labeled training examples, producing three outputs: the *command* type, the *DSR* type, and the *static spatial relation* (if available). The command type is domain-specific, and may include commands such as: robot movement, object movement, learned tasks, etc. The resulting specifications are then passed on to the planning module to search for appropriate robot task solutions.

For more information regarding our approach to modeling spatial relations and our HRI framework design, we refer the reader to [7, 8, 9].

B. Probabilistic Extraction of Instruction Sequences

In our prior work we utilized a phrase structure grammar capable of parsing spatial language directives that instructed a variety of robot tasks, including for example, robot movement commands (e.g., "Go inside the kitchen"), object manipulation/placement commands (e.g., "Put the book on top of the coffee table"), and spatial commands without explicit prepositions (e.g., "Leave the room"). Table I displays the basic rules of this grammar for illustration purposes; the complete grammar is slightly more complex [8]. As shown, the non-terminal symbols defined by the constituency rules include those for sentences (S), noun phrases (NP), and terminating noun phrases (N').

While the grammar presented is capable of capturing many different types of spatial language instructions (including those with hierarchical noun phrases) provided as

TABLE II. PROBABILISTIC INSTRUCTION SEQUENCE EXTRACTION PROCEDURE EXAMPLE WITH ITERATION NOTES

Word Index	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
Word	"PR2	please	go	into	my	room	and	get	me	my	shoes	thank	you"
POS Tag	N	V	V	P	PRP$	N	CC	V	PRP	PRP$	N	V	PRP
Non-Terminals	NP$_1$	-	S$_4$	-	NP$_2$	NP$_1$	-	S$_4$ S$_2$	NP$_1$	NP$_2$	NP$_1$	S$_2$	NP$_1$
Algorithm Steps	*No S*	*No S*	*Valid S Found len=4*	*(skip)*	*(skip)*	*(skip)*	*No S*	*Valid S Found len=4*	*(skip)*	*(skip)*	*(skip)*	*Invalid S*	*No S*

Note. Input text = "PR2 please go into my room and get me my shoes thank you". Parser output is shown along with algorithm steps during iteration over the word indices of the POS tag assignment array. Non-terminal symbol lengths are shown in subscripts. Sentences are validated against the context of environment and the inferred command semantics/requirements. Final instruction sequence output by algorithm: {go into my room, get me my shoes}

TABLE I. GRAMMAR FOR SPATIAL LANGUAGE DIRECTIVES

S → V P* NP	NP → N'
S → V NP P* NP	NP → N' P NP
N' → (Det) A* N+	NP → NP and NP

Note. POS Tags: V = Verb, P = Preposition, N = Noun, A = Adjective, Det = Determiner

discrete input, its scope is limited when applied to natural language input taken as a whole. Specifically, it is unable to parse the many non-spatial phrases that users often employ when providing instructions through natural language. In addition, the grammar only allows for a single instruction per sentence, yet in practice, people often sequence multiple instructions together within a single utterance that must therefore be appropriately segmented.

To address these limitations in practice with end users, we have developed a probabilistic parsing procedure capable of extracting a sequence of grammatical instructions (partial parses) from unconstrained natural language input for subsequent robot task planning and execution. Following is an overview of the five steps of the algorithm:

1) Part-of-Speech (POS) Tag Assignment: The first step of the algorithm is to determine the POS tags (terminals) for each word of the input text. We use the Stanford NLP Parser [14] to generate default POS tags; however, because the Stanford parser does not have access to situational context, it occasionally assigns POS tags incorrectly (e.g., "Place" assigned as a noun (N), instead of as a verb (V)). To address this issue, we additionally apply domain-specific POS tags taken from a pre-defined lexicon when there is a disagreement with the default tags, so that the parser may consider both tag options. As a result, there may be multiple assignments generated for a given input text (with exponential growth). In practice, however, there are typically only 1-4 tag assignment arrays for each input, and invalid tag assignments are discarded quickly in the following step due to grammatical incorrectness. For each of the generated tag assignment arrays the algorithm performs steps 2-4; the procedure then concludes with step 5.

2) Parse Word/POS Tag Array using Grammar: Given a word/POS tag assignment array, the algorithm proceeds to extract the corresponding high-level tags (non-terminals) for the input as defined by the constituency rules of the grammar (see Table I). The result is that for each word position, there exists a set of non-terminal symbols parsed by the grammar that begin at that index, each with an associated length corresponding to the number of consecutive terminal symbols that serve as constituents for the non-terminal symbol. From this representation, the algorithm only considers non-terminal symbols that denote grammatical sentences (i.e., S).

3) Find Maximum Probable Sentences: For each word index of the word/POS tag assignment array, all sentences (symbol S) with maximum length among the available sentences are tested for validity within the context of the environment and the semantics of the inferred instruction. The algorithm only considers sentences of maximum length among those available as a heuristic to avoid evaluating partial sentences unnecessarily. If no sentences exist at the current index, the algorithm moves on to the next index.

Valid sentences are those whose NPs can be grounded uniquely in the world, and whose parameters meet the specifications of the inferred command. An example error would be if the inferred command was [Object Movement] and the grounded NP parameter was [*the kitchen*], as [*the kitchen*] is of type [Room] and hence not movable by the robot. In this case a flag would be thrown and the sentence would be deemed invalid. This validation procedure is discussed in detail in [8].

Among the sentences at the current index found to be contextually and semantically valid, the sentence of maximum probability (calculated during the inference process) is chosen as the most likely sentence found at the current index, and the algorithm then skips the word indices covered by the sentence and continues on searching for valid sentences at the next available index.

4) Form Instruction Sequence Candidate: All of the valid sentences found within the word/POS tag assignment array (i.e., those with maximum probability at their respective word positions) are then combined to form the optimal instruction sequence candidate for the specific POS tag assignment of the natural language input.

5) Find Maximum Probable Instruction Sequence: Once all instruction sequence candidates are gathered, the final instruction sequence returned by the algorithm is that which is of maximum probability among the candidates (determined by multiplying together each of the probabilities of the individual sentences in the sequence). To allow for fair

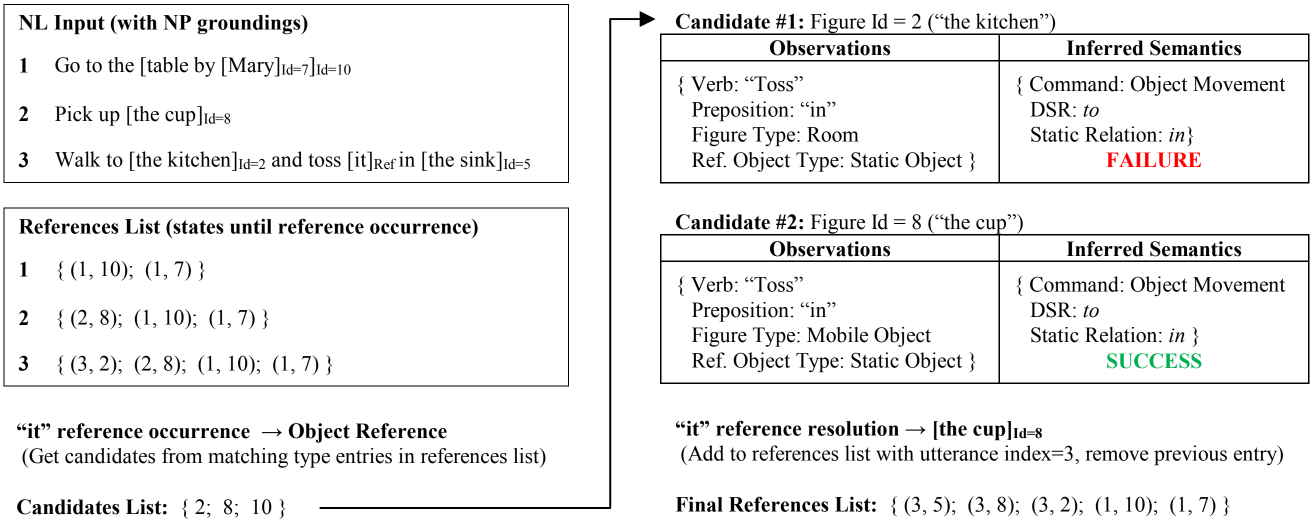

NL Input (with NP groundings)
1 Go to the [table by [Mary]$_{Id=7}$]$_{Id=10}$
2 Pick up [the cup]$_{Id=8}$
3 Walk to [the kitchen]$_{Id=2}$ and toss [it]$_{Ref}$ in [the sink]$_{Id=5}$
References List (states until reference occurrence)
1 { (1, 10); (1, 7) }
2 { (2, 8); (1, 10); (1, 7) }
3 { (3, 2); (2, 8); (1, 10); (1, 7) }
"it" reference occurrence → Object Reference
(Get candidates from matching type entries in references list)
Candidates List: { 2; 8; 10 }
Candidate #1: Figure Id = 2 ("the kitchen")
Observations
Inferred Semantics
{ Verb: "Toss" Preposition: "in" Figure Type: Room Ref. Object Type: Static Object }
{ Command: Object Movement DSR: to Static Relation: in}
FAILURE
Candidate #2: Figure Id = 8 ("the cup")
Observations
Inferred Semantics
{ Verb: "Toss" Preposition: "in" Figure Type: Mobile Object Ref. Object Type: Static Object }
{ Command: Object Movement DSR: to Static Relation: in }
SUCCESS
"it" reference resolution → [the cup]$_{Id=8}$
(Add to references list with utterance index=3, remove previous entry)
Final References List: { (3, 5); (3, 8); (3, 2); (1, 10); (1, 7) }

Figure 2. Reference resolution example for the instruction "toss it in the sink" expressed by the user during spatial language discourse. Parsed NPs of the natural language input are shown in brackets with their corresponding unique grounding ID numbers as subscripts.

comparison, candidates are evaluated only against others of equal length (number of sentences), and instruction sequences of greater length are favored.

Table II illustrates the probabilistic instruction sequence extraction procedure with an example by displaying the word/POS tag assignment array for the input "PR2 please go into my room and get me my shoes thank you", along with the corresponding parsed non-terminal symbols and resulting algorithm steps. In the example, the procedure finds the following instruction sequence most likely given the natural language input: {Go into my room, Get me my shoes}.

C. *Reference Resolution*

In natural language discourse, people often refer to entities, or groundings, which have been previously mentioned or discussed through the use of *anaphora*. Examples include references to objects (e.g., "it", "itself", "this", "that") and people (e.g., "he/she", "him/her", "him/herself"). In addition, anaphoric expressions typically refer to an entity introduced by a noun phrase within a recent utterance in the discourse history (usually within one or two past utterances) [20]. The prevalence of anaphora in natural language discourse necessitates a computational approach for resolving such references for use in real world human-robot interaction scenarios with non-expert users.

In this subsection, we present our approach to resolving anaphoric references to both objects and people in the context of user-guided spatial language discourse. Our reference resolution procedure is similar in nature to those that have been developed previously based on related principles [20, 21], albeit with the distinction of its optimization for use within the framework of our spatial language architecture, and in particular, for its designed integration with our probabilistic instruction sequence extraction procedure (presented in the previous subsection).

At a high level, our procedure for resolving anaphoric references within user discourse can be summarized by the following key concepts: 1) entities represented in the world (e.g., mobile objects, static objects, rooms, people, etc.) are associated internally with numerical identifiers that enable unique identification during the NP grounding process; 2) as these groundings are referenced in the discourse (usually by name) their unique grounding ID numbers are added to a global list of recent references; and 3) upon encountering anaphoric expressions within the discourse, the groundings in the recent references list are used as candidate references in an attempt to uniquely resolve the referential expression to the specific grounding that the user intended to convey.

More specifically, in our approach anaphoric expressions are categorized as either Object References or Human References, depending on whether or not the anaphor encountered refers to a person. In addition, anaphoric references to persons are further categorized by gender (male/female). When adding groundings to the global list of recent references, an entry pair is made with both the current utterance index and the grounding ID. If a prior entry is found with the same grounding ID, it is removed in favor of the new entry. The utterance index is incremented after every utterance spoken during discourse, and it is included in the global list to enforce the consideration of only the references expressed within the most recent utterances (in our implementation we utilize a history size of three utterances).

In resolving an anaphoric expression, only recent reference groundings with matching type (object vs. human, male vs. female) are allowed as candidates, and the list of candidate groundings is prioritized with the most recent references at the top. During the grounding process, child NPs that can be directly grounded (i.e., not anaphoric) are added to the current list of recent references; alternatively, child NPs that contain anaphora instead merge their candidates list with those of sibling NPs to form one combined candidates list for the parent NP. Once all child NPs are processed and either the figure and/or reference object NP parameters of the spatial language instruction contain anaphora, the command semantics are evaluated for each of the possible candidate groundings until the first successful assignment is found. This greedy approach to resolving the reference is reasonable under the assumption that the list of candidates is ordered with the most likely candidates on top (the most recent are set as a best estimate).

As previously mentioned, our procedure for anaphora resolution was designed to be well integrated with our probabilistic instruction sequence extraction procedure. This integration is actually a crucial necessity, as determining the optimal instruction sequence for a given utterance containing anaphora depends entirely on accurate reference resolution. Furthermore, if multiple POS tag assignments exist for the given utterance, separate reference lists must be concurrently maintained and adjusted according to the evolving context of the different threads of possible discourse under consideration. Yet, in practice, the integration is seamless: each POS tag assignment is given its own references list that originally is a copy of the most current global references list (most current before utterance processing began). Additionally, in step 3 a temporary list is used for each new sentence grounding and validation check, which is set initially to the most recent references list encapsulating the instructions (of maximum probability) that have already been accepted for the current POS tag assignment's instruction sequence. Lastly, the resulting references list for each of the candidate instruction sequences is stored until final determination of the maximum probable sequence, whose corresponding references list then becomes the global list.

Fig. 2 illustrates the reference resolution procedure with an example discourse scenario, displaying the spatial language input and the evolving state of the global references list, among other properties of the algorithm.

IV. Pragmatics For Interaction with People and Generalized Transfer to Robot System

A. *Pragmatics for Interaction with People*

Our prior work has demonstrated that the A* search algorithm can be used effectively in conjunction with the semantic field model of spatial prepositions to generate robot task solution plans for execution of spatial language instructions provided by the user, including under user-specified natural language constraints [7, 8, 9]. However, the approach was tested only in simulated 2D/3D environments and without modeling direct interaction with people. In this section we will discuss pragmatic considerations in transferring our approach to physical robots for interactions with people in real world environments, and how each was applied in our methodology towards enabling natural human-robot interaction.

Safety is perhaps the most important pragmatic constraint to consider when designing robot systems that are to interact with people. When generating robot task solution plans for given user instructions, it is important that the path/actions taken by the robot be safe for the user, but also for the robot. In our prior work the robot task solution plans were generated to achieve optimality in terms of both distance traveled and adherence to user defined constraints, without consideration for the value of generating "safe" solution paths. To address this issue, we incorporated specific pragmatic safety fields into the planning process, one for the robot and another for people within the environment.

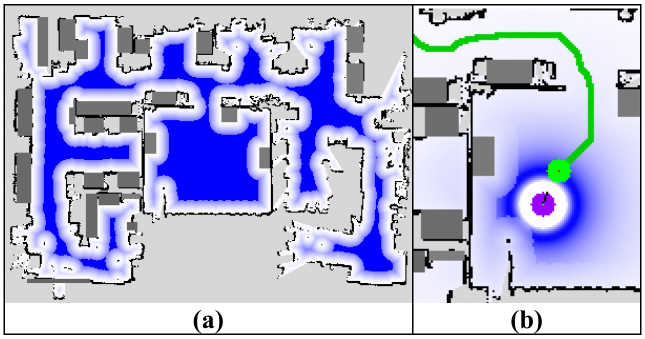


Figure 3. (a) SLAM map of laboratory space with pragmatic field for robot safety shown; (b) Example robot approach behavior with combined semantic/pragmatic field shown for at/person safety.

Our prior work has demonstrated the ease of incorporating pragmatic constraints in our methodology with the use of spatial pragmatic fields [8]. These fields have the same representation as the semantic fields used for computing spatial relations within the environment, and can easily be combined for generating robot task plans that consider both the semantics and the pragmatics of the given instruction. The safety field for the robot was generated using a Gaussian function and a safety threshold specifying the minimum desired distance from obstacles and also the function mean (set to 2*robot radius). Field values for points in the world were designated based on their distance to the nearest obstacle, where distances above the threshold would result in a maximal field value (1.0) and distances below the threshold would be set according to the Gaussian. The safety field for people is generated similarly, with the distance parameter instead referring to the distance from the person. Fig. 3(a) shows an example robot safety field computed for a real world laboratory environment using a SLAM map, and Fig. 3(b) illustrates use of the person safety field (merged with the *at* semantic field and pragmatic field for robot safety) in a robot solution for the instruction "Come to me".

The resulting pragmatic fields have been integrated into the A* cost function of our planning procedure, so as to designate preference for safer solution paths for both the robot and people. Other pragmatic fields can easily be incorporated during planning using our methodology, including for example, those that enforce appropriate approach behaviors (e.g., not from behind) and person-to-person interaction spaces (e.g., do not cross) [19].

B. *Generalized Transfer to Robot System*

In order to successfully transfer our spatial language interpretation framework to a physical robot system, there are a few technical challenges that first need to be addressed. The primary of which is a procedure for the translation of the discretized path returned by our planner into appropriate robot motor commands (e.g., wheel velocities) that result in the robot following the desired path. Another major consideration is the autonomous generation of a map of the environment to be used during planning. Last but not least,

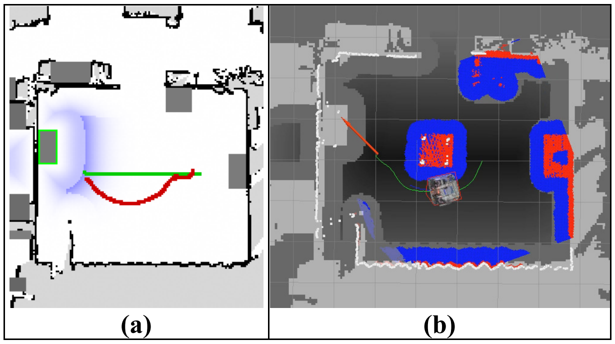


Figure 4. Dynamic obstacle avoidance for instruction "Go to the dinner table" (a) Planned path (green) and actual path (red); (b) Visualization of obstacles detected in robot's local map and global plan after robot re-planning.

the robot must be able to localize itself within the generated map of the environment using onboard sensors. Fortunately, previous work conducted by researchers in the field (e.g., [17]) have already designed solutions to these challenging research problems, many of which have been packaged within the software framework of the Robot Operating System (ROS) [18] and which are freely available for use. In transferring our approach to a physical robot system (the PR2 robot platform) we utilized the ROS software packages available for the generation of SLAM maps (gmapping), robot localization (AMCL), and robot navigation planning (global + local planning using DWA).

Translation from the discretized plan to robot motor commands was accomplished by creating a cost map for the ROS navigation package to use during planning that would strongly favor points along (or close to) the planned solution path. Once created, the cost map is sent to the ROS navigation stack along with the desired goal position in the map. The result is a smooth path that takes into account the motion model of the robot (e.g., omnidirectional vs. differential drive) while following very closely to the path generated by our spatial language-based HRI framework (DWA parameters used: path bias = 30, goal bias = 10). The navigation also takes into account local obstacles encountered during task execution, and is able to quickly re-plan upon encountering an obstruction. Fig. 4 shows an example of dynamic obstacle avoidance during task execution for obstacles not found in the static map, displaying actual data from a test run with the PR2 robot where a table was introduced into the environment not present in the static map. Fig. 4(b) additionally shows the cost map that was generated for task planning (shown in grayscale with values scaling linearly with distance to the original discretized path produced by our spatial planner).

By utilizing the ROS framework to abstract away the generation of robot motor commands from the spatial language task solution, the transfer process is generalized and can easily be replicated for a variety of robot platforms.

V. Evaluation

To evaluate the ability of our robot system to follow natural language directives involving spatial language, we first analyzed the performance of the physical robot platform (PR2 robot) at reaching the desired destination specified in the spatial language instruction. To evaluate the robot's performance, we conducted a spatial positioning task experiment that consisted of instructing the robot to move to a desired location satisfying a given spatial relation with respect to one or more groundings in the environment, expressed through natural language. The experiment consisted of 14 test instructions given to the robot, two for each static spatial relation analyzed (*near*, *away from*, *between*, *inside*, *outside*, *at*) and two additional test runs for the spatial relation *at*, which is associated with the most common path preposition utilized in spatial instruction tasks ("to"). After each experiment run, the end location of the robot was measured against the goal position generated by our spatial language interpretation framework; specifically the distance between the target end point and actual robot end point was recorded. An example run of the experiment is shown in Fig. 5 for the instruction "Stand between the printer desk and the whiteboard", displaying both the planned path and actual path taken by the robot during task execution.

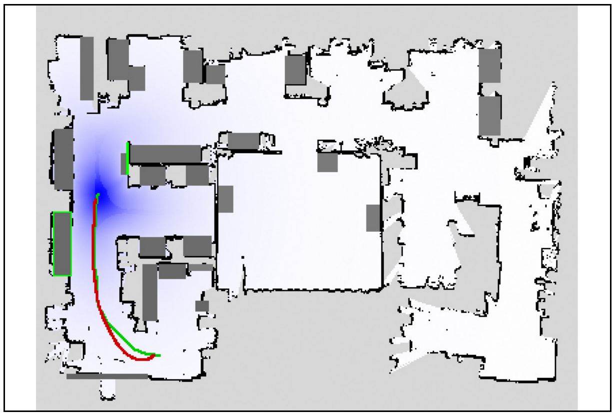


Figure 5. Combined semantic/pragmatic field and execution result for the task "Stand between the printer desk and the whiteboard"

TABLE III. Accuracy of Final Robot Positions in Spatial Task Experiment

Measure	Mean (Std.)
Distance to Goal Position	0.19 m (0.037 m)
Distance to AMCL Position Estimate	0.07 m (0.038 m)

Note. Distances calculated from the actual robot position measured after task completion

Table III shows the results of the analysis, which demonstrate the robot's notable accuracy in estimating its position within the environment, as the distance errors were very small (within 0.2 m). This amount of distance between the final point of the robot and the planned goal point is to be expected, as the ROS navigation module operated with an acceptable goal distance threshold of exactly 0.2 m. The minor differences observed between the robot's AMCL position estimates and the actual final position highlight the effectiveness of the robot's onboard localization procedure (implemented in ROS), and demonstrate the ease of which the robot was able to follow the spatial language instructions provided to it by our framework in practice in a real world environment.

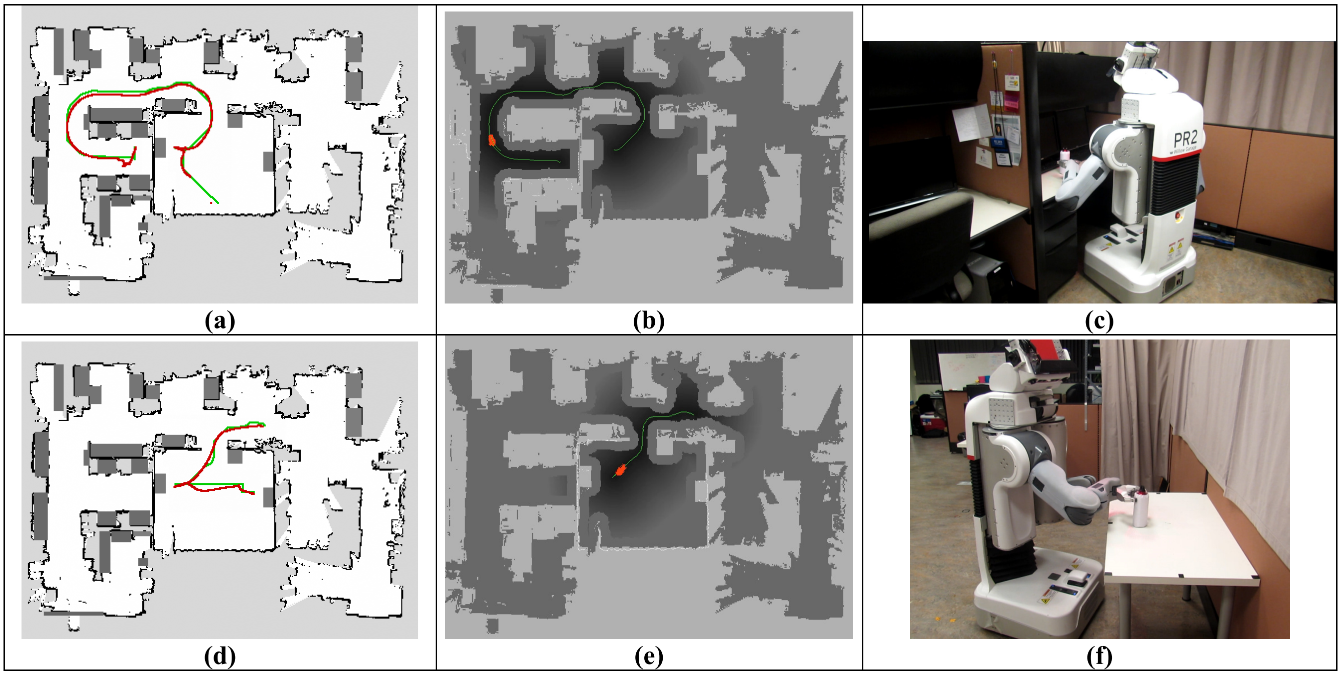


Figure 6. (From left to right) Planned (green) and executed paths (red), cost map used for navigation planning with AMCL particles corresponding to robot position estimates, and photograph of PR2 robot just before task termination for test runs 1-2. (a),(b),(c) run 1; (d),(e),(f) run 2.

Next, to demonstrate the capabilities of 1) our probabilistic instruction sequence extraction procedure, 2) our approach to resolving anaphoric expressions, and 3) our integration of pragmatic constraints involving safety fields for interacting with and operating in environments together with humans, we ran three additional test runs of our spatial language discourse interpretation and HRI framework.

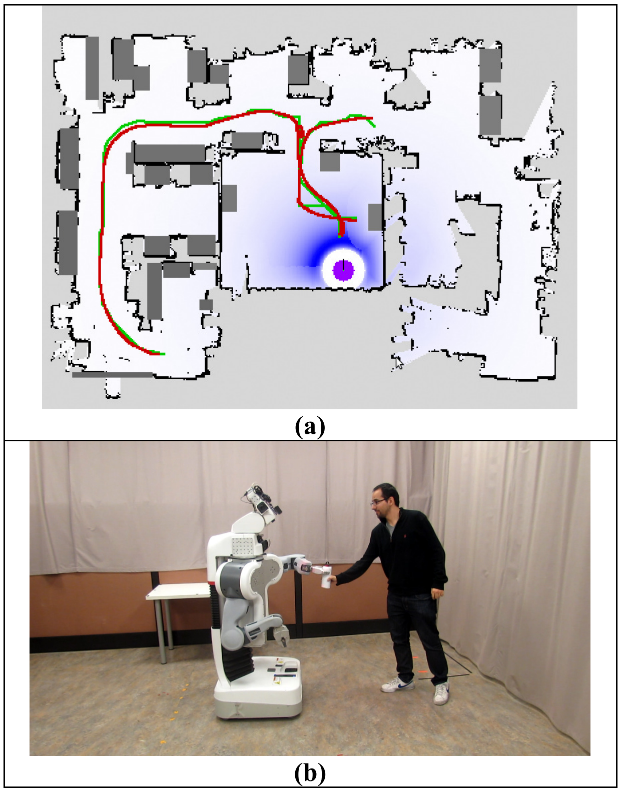
Each test run involved the user engaging the robot in spatial language discourse, and in particular, providing a series of instructions, with and without the use of anaphora, for the robot to track, resolve references, and execute appropriate task solutions. In total, 11 instructions were evaluated. The spatial language discourse provided to the robot in each of the three test runs are shown in Table IV, together with the instruction sequences inferred by our probabilistic instruction sequence extraction procedure. Natural language input given to the robot included those with multiple instructions within a single utterance, which could also contain non-spatial language (e.g., "PR2 can you please head to the dinner table and then pick up the water bottle and take it to my desk so I can have a drink later"), and those with multiple anaphora ("Put it on top of it") for the robot to attempt to interpret within the context of the discourse.

TABLE IV. INSTRUCTIONS GIVEN IN TEST RUNS 1-3 WITH INFERENCE RESULTS FOR INSTRUCTION SEQUENCES

Run	Natural Language Instructions
1	"PR2 can you please head to the dinner table and then pick up the water bottle and take it to my desk so I can have a drink later" Head to the dinner table Pick up the water bottle Take it to my desk
2	"Go ahead and grab the cup if you can and then it would be great if you could go to the kitchen counter and put it on top of it for me" Grab the cup Go to the kitchen counter Put it on top of it
3	"Come into the pen" Come into the pen "Lift up the object close to Juan" Lift up the object close to Juan "Give him it and then step back outside the pen and wait by the entryway" Give him it Step back outside the pen Wait by the entryway

Note. Distinct utterances are listed on separate lines. Instruction sequences inferred by the probabilistic extraction procedure are shown in red

Fig. 6 and Fig. 7 illustrate the performance of the robot during each of the test runs. As evidenced by the results, the robot was able to successfully perform all of the tasks requested by the user in each of the natural language instructions of the test runs. Notable results include the ability of the robot to resolve the multiple anaphoric references in the instruction "Put it on top of it" during the second test run. In this instance, the robot correctly resolved the first reference to the grounding of [the cup], mentioned in the first utterance of the discourse, after disqualifying the initial candidate (most recently grounded NP) of [the kitchen counter] as invalid semantically due to inconsistencies with the parameter requirements of the inferred command of [Object Movement]. Similarly, the instruction "Give him it" expressed by the user during the third test run was correctly resolved by the robot in accordance with the context of the spatial language discourse ("him" → [Juan], "it" → [the object close to Juan]). Fig. 7(b) displays the interaction between the robot and the user at the time of object transfer as performed by the robot during execution of this task.


(a)

(b)

Figure 7. Test run 3 results (a) Planned path (green) and actual path (red) with semantic/pragmatic fields calculated for hand-off behavior; (b) PR2 robot handing bottle (grounded object referent) to intended person (ground referent for "him") during task execution.

The capability of the robot in successfully interpreting the spatial language discourse expressed during each of the test runs, while also taking into account the pragmatics of the interaction, demonstrates the potential of our approach for use in real world environments with target users.

VI. Conclusion

We have described the need for enabling autonomous service robots with spatial language understanding and discourse modeling to facilitate natural communication with non-expert users for task instruction and anaphoric reference resolution, and have presented a general approach we have developed toward addressing this research challenge. The results obtained from our evaluation testing demonstrate the potential of our methodology for representing dynamic spatial relations, grounding and interpreting the semantics of natural language instructions probabilistically, extracting instruction sequences from unconstrained natural language input, and resolving anaphoric expressions within the context of the current discourse with the user.

Acknowledgment

We are grateful to Yixin Cai and Han Xiao for their help with ROS integration, PR2 software development, and in collecting experiment data. We also thank Carlos A. Fasola for providing linguistics expertise, numerous helpful suggestions, and insightful feedback regarding this work.

References

[1] B. Landau and R. Jackendoff. What and where in spatial language and spatial cognition. *Behavioral and Brain Sciences*, 16(2):217–265, 1993.

[2] J. O'Keefe. Vector grammar, places, and the functional role of the spatial prepositions in English, E. van der Zee and J. Slack, Eds. Oxford: Oxford University Press, 2003.

[3] S. Tellex, T. Kollar, S. Dickerson, M. R. Walter, A. G. Banerjee, S. Teller, and N. Roy. Approaching the Symbol Grounding Problem with Probabilistic Graphical Models. *AI Magazine*. 32(4): 64-76, 2011.

[4] T. Kollar, S. Tellex, D. Roy, and N. Roy. Toward Understanding Natural Language Directions. In Proc. ACM/IEEE Int'l Conf. on Human-Robot Interaction (HRI), 259–266, 2010.

[5] Y. Sandamirskaya, J. Lipinski, I. Iossifidis, and G. Schöner. Natural human-robot interaction through spatial language: a dynamic neural field approach. 19th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), pp. 600-607, 2010.

[6] M. Skubic, D. Perzanowski, S. Blisard, A. Schultz, W. Adams, M. Bugajska, and D. Brock. Spatial Language for Human-Robot Dialogs. *IEEE Transactions on SMC Part C*, Special Issue on Human-Robot Interaction, 34(2):154-167, 2004.

[7] J. Fasola and M. J. Matarić. Modeling Dynamic Spatial Relations with Global Properties for Natural Language-Based Human-Robot Interaction. In Proc. IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), Korea, Aug 2013.

[8] J. Fasola and M. J. Matarić. "Using Spatial Semantic and Pragmatic Fields to Interpret Natural Language Pick-and-Place Instructions for a Mobile Service Robot". In International Conference on Social Robotics (ICSR), Bristol, UK, Oct 2013.

[9] J. Fasola and M. J. Matarić. "Using Semantic Fields to Model Dynamic Spatial Relations in a Robot Architecture for Natural Language Instruction of Service Robots". In IEEE/RSJ International Conference on Intelligent Robots and Systems, Japan, Nov 2013.

[10] P.E. Rybski, J. Stolarz, K. Yoon, and M. Veloso. Using dialog and human observations to dictate tasks to a learning robot assistant. *Journal of Intelligent Service Robots*, 1:159-167, 2008.

[11] H. Kress-Gazit, G.E. Fainekos, and G.J. Pappas. Translating structured English to robot controllers. *Advanced Robotics*, 22, 1343–1359, 2008.

[12] J.D. Kelleher, F.J. Costello. Applying Computational Models of Spatial Prepositions to Visually Situated Dialog. *Computational Linguistics*, 35(2):271-306, 2009.

[13] L.A. Carlson, and P.L. Hill. Formulating spatial descriptions across various dialogue contexts. In K. Coventry, T. Tenbrink, & J. Bateman (Eds.), Spatial Language and Dialogue, 88-103. New York, NY: Oxford University Press Inc, 2009.

[14] D. Klein, and C.D. Manning. Accurate Unlexicalized Parsing.In Proceedings of the 41st Meeting of the Association for Computational Linguistics, pp. 423-430, 2003.

[15] C. Matuszek, E. Herbst, L. Zettlemoyer, and D. Fox. Learning to Parse Natural Language Commands to a Robot Control System. In Proc. of the International Symposium on Experimental Robotics (ISER), Québec City, Canada, 2012.

[16] R. Cantrell, P. Schermerhorn, M. Scheutz. Learning actions from human-robot dialogues. In Proc. IEEE RO-MAN, pp.125-130, 2011.

[17] D. Fox, W. Burgard, and S. Thrun. The dynamic window approach to collision avoidance. *IEEE Robotics and Automation*, 4(1), 1997.

[18] Robot Operating System (ROS). http://www.ros.org

[19] Dizan Vasquez, Procopio Stein, Jorge Rios-Martinez, Arturo Escobedo, Anne Spalanzani, Christian Laugier: Human Aware Navigation for Assistive Robotics. ISER. 449-462, 2012.

[20] Daniel Jurafsky and James H. Martin. Speech and Language Processing: An Introduction to Natural Language Processing, Speech Recognition, and Computational Linguistics. Prentice-Hall, 2008.

[21] Jaime G. Carbonell and Ralf D. Brown. Anaphora resolution: a multi-strategy approach. In *Proceedings of the 12th conference on Computational linguistics - Volume 1* (COLING '88), Dénes Vargha (Ed.), Vol. 1., 96-101, 1988.