POWER: A Domain-Independent Algorithm for Probabilistic, Open-World Entity Resolution

Tom Williams and Matthias Scheutz

***Abstract*— The problem of uniquely identifying an entity described in natural language, known as *reference resolution*, has become recognized as a critical problem for the field of robotics, as it is necessary in order for robots to be able to discuss, reason about, or perform actions involving any people, locations, or objects in their environments. However, most existing algorithms for reference resolution are domain-specific and limited to environments assumed to be known *a priori*. In this paper we present an algorithm for reference resolution which is both domain independent and designed to operate in an open world. We call this algorithm POWER: Probabilistic Open-World Entity Resolution. We then present the results of an empirical study demonstrating the success of POWER both in properly identifying the referents of referential expressions and in properly modifying the world model based on such expressions.**

I. Introduction

A key ability for robots designed to interact with humans is the ability to identify those entities referenced by human interlocutors in natural language (NL); if a robot is to discuss, interact with, or travel to some object or location referenced by a human, the robot should have some notion of the identity of that object or location.

There have been many recent approaches to this problem, known as *reference resolution* or *language grounding* (e.g., [1], [2], [3]). However, the majority of these approaches operate under tight domain constraints and an assumption of full environmental knowledge. We believe it both necessary and possible to move beyond these constraints.

First, we believe that algorithms for understanding referential expressions should be domain-independent. As a robot may need to resolve expressions which simultaneously refer to people, places, and objects, any algorithm it uses to perform reference resolution should be domain independent in nature. This entails restricting the function of such an algorithm to guiding the search through the space of possible candidate referents, and leaving the task of "constraint evaluation" (e.g., evaluating whether two rooms are "across" from each other) to separate domain-dependent processes. This functional organization will also allow reference resolution and constraint evaluation mechanisms to be developed independently, and easily integrated together.

Second, a robot must be able to handle not only the *uncertainty* of its own knowledge, but the *incompleteness* of that knowledge, as robots may need to operate in environments where they are not familiar with every entity which could conceivably be referenced in conversation. Consider a robot which, upon entering a building for the first time, is told by an interlocutor "My office is down the hall across from the kitchen." That robot should be able to discuss and reason about that location without knowing its precise location and without having visited it, and should be able to find its way to its interlocutor's office from any other part of the building if it later needs to travel to it.

The problem of open-world reference resolution produces two interesting challenges: (1) determining which parts of a referential expression refer to known vs. unknown entities, and (2) determining how to modify a robot's world model or knowledge base in response to new information about entities either known or unknown. In previous research, we introduced novel algorithms for *spatial* (i.e., location-based) reference resolution in an *open world* (i.e., one in which the entire environment is not known *a priori*) [3] which made first steps towards tackling these two challenges. However, those algorithms were neither probabilistic nor domain-independent in nature.

In this paper, we present a reference resolution algorithm that (1) handles uncertain knowledge, (2) improves on the open-world reference resolution techniques used in our previous work, and (3) generalizes to handle resolution of any entity. We call this the **POWER** algorithm, since the combination of these abilities allows for **P**robabilistic **O**pen-**W**orld **E**ntity **R**esolution.

After discussing previous approaches and presenting our algorithm, we describe the results of an empirical study demonstrating the success of POWER both in properly identifying the referents of referential expressions and in properly modifying the world model based on such expressions.

II. Previous Work

In this section, we will discuss previously presented algorithms for reference resolution, paying special attention to whether each algorithm (1) is able to deal with uncertain knowledge, (2) operates under open or closed world assumptions, and (3) (if it operates under open world assumptions) properly modifies its world model in response to new information. As a vast number of domain-specific approaches have been presented, we only examine those which deal with uncertain or incomplete knowledge.

Matuszek et al. [2] used Statistical Machine Translation techniques to translate NL directions into routes through an open world. Information was then added to the robot's world model as it executed such a route, but no information was added to the world model from the utterance directly, thus

Tom Williams and Matthias Scheutz are with the Human-Robot Interaction Laboratory at Tufts University, Medford, MA, USA. (williams, mscheutz)@cs.tufts.edu

preventing the robot from discussing or reasoning about mentioned locations without visiting them first. This approach did not handle uncertain knowledge.

In previous work [3], we presented *SPEX: The Spatial Expert*, which added new topological locations to its world model when an utterance did not seem to refer to a known location. This approach treated route planning as a process separate from spatial reference resolution, thus allowing a robot to discuss hypothetical locations without having visited them. However, it did not handle uncertain knowledge.

Fasola and Mataric present *Semantic Fields* [4], an approach that represents spatial relations as probability density functions over points in a known metric map (e.g., "near" is a function that ascribes higher probability to points closer to the object in question). Resolution is performed by comparing nouns against a label knowledge base. If a single match is found, the phrase is considered resolved. If no matches are found, the user is prompted for clarifying information. If multiple matches are found, the algorithm looks to the noun's parent clause, and uses its attached prepositional phrase (if any) to disambiguate by choosing the candidate with the highest semantic field score. While this approach handles uncertain knowledge in that it finds the most likely among multiple candidates, its use of a closed world means that those semantic fields are not used to assess the appropriateness of a candidate if it is the only one found. Furthermore, this assumption of a closed world means that the approach would not be able to resolve references to as-yet unobserved locations or objects.

Kruijff et al. present an accident investigation robot which incrementally builds a hierarchical topological map of the accident scene [5]. While the robot uses a probabilistic approach which makes use of the uncertainty of its own knowledge, it is not able to handle open-world aspects when performing reference resolution.

Chai et al. present a greedy algorithm for probabilistic reference resolution in multi-modal user interfaces [6]. This algorithm is informed by several linguistic theories (i.e., *Conversation Implicature* and the *Givenness Hierarchy*) and allows recognized gestures to bias resolution. However, as the algorithm is intended for use in a user interface in which the only candidate referents are those which appear on the screen, it does not attempt to handle open worlds.

Kollar, Tellex et al. present the *Generalized Grounding Graph* or G^3 approach [7], [8]. The G^3 approach dynamically instantiates probabilistic graphical models based on the structure of incoming NL utterances. It then performs a beam search through an initial domain of salient objects and locations (with variables traversed in reverse order of nesting depth) to find the set of referents that most probably satisfy the produced PGM. Whether a place or set of places satisfy a particular spatial relation is learned from large labeled corpora. While G^3 allows new locations to be learned through exploration, it does not allow new locations to be learned through *dialogue*, and thus like Matuszek et al.'s approach, G^3 does not allow a robot to discuss or reason about new places without first visiting them.

Duvallet et al. present an approach which samples learned models of spatial relations (e.g., "behind") to create routes which will likely lead the robot to unknown entities [1]. However, like other discussed approaches, commands are parsed directly to actions, and thus this approach does not appear to allow a robot to discuss or reason about new places without first visiting them. Furthermore, this approach can only resolve references to unknown objects if they are described in reference to objects whose identities are provided *a priori*. The ideal algorithm would be able to handle worlds in which the identities and properties of known objects and locations can be uncertain, and in which unknown entities may be described with respect to other unknown entities.

III. Algorithm

We now present the **POWER** algorithm for **P**robabilistic **O**pen-**W**orld **E**ntity **R**esolution.

POWER takes five parameters:

1) S_0 and S_1: initially identical sets of semantic constraints (represented as formulae such as $in(X, Y)$ or $color(Z, green)$) imposed by the referential expression to be resolved. These constraints may be produced by a semantic parser such as TLDL [9] or Mink [10].
   While S_1 will change each time the algorithm recurses, S_0 will remain the same, so that the original set of constraints will still be available at each level of recursive depth.
2) C: a *Consultant* for the target domain (e.g., a spatial reasoning, vision processing, or belief monitoring component). This Consultant provides an interface to the knowledge base or world model M of its target domain. World model M may have any sort of storage and retrieval mechanisms (for example, if the *spatial* domain is targeted, M might be a metric-topological map, with information retrieved using some mathematical process, such as semantic fields [4]), but it is assumed to contain some number of unique atomic *entities* (e.g., discrete locations).
   Consultant C has three capabilities: (1) provision of the initial domain of candidate atomic entities for a given variable, (2) calculation of the probability that a particular configuration of entities satisfy a given property, and (3) modification of M based on information provided through natural language.
   For example, a spatial reasoning component may be able to (1) provide a list of places likely to be referenced in the current context, (2) calculate the likelihood that a particular place can be considered a hallway, and (3) add new places to its map based on the utterance "I'll meet you in the kitchen that's at the end of the hall".
3) V: an ordered list of variables contained in S_1, in ascending order of certainty that the robot's interlocutor believes the robot to be familiar with the entity corresponding to that variable. For example, in the phrase "the room at the end of the hall", the room is being described in relation to the hall, so it is

reasonable to assume that the interlocutor thinks the robot will be more familiar with the hall than with the room, and thus the variable associated with the room will be listed before the variable associated with the hall. This is a critical assumption of our approach. POWER distinguishes between old and new information by attempting to find the first satisfactory cut of V that divides it into a *prefix* of new entities and a *suffix* of existing entities. It is thus critical that such a cut is likely to exist in the ordering chosen for V. There may be multiple methods for choosing an ordering of V which lead to satisfactory cuts, but we do so by arranging the variables contained in the query in reverse order of nesting depth.

4) H: an (initially empty) priority queue of hypotheses.

Given these parameters, **POWER** (as seen in Algorithm 1) performs a best-first search for the most probable satisfaction of the variables contained in S_0, recursing when no probable hypothesis can be found.

Algorithm 1 POWER(S_0, S_1, C, V, H)

```
1: V: ordering of variables found in S_1
2: S_0: set of formulae
3: S_1: set of formulae
4: C: consultant
5: H: an initially empty priority queue
6: A = ∅ (the set of solutions)
7: if H = ∅ then
8:     v = first_unbound_variable(S_1.head)
9:     for all c ∈ C.initial_domain do
10:        H.enqueue({{v → c}, S_1.clone, 1.0})
11:    end for
12: end if
13: while H ≠ ∅ do
14:    h = H.pop
15:    s = h.constraints.pop
16:    if (∃v ∈ s.vars | v ∉ h.bindings) then
17:        for all c ∈ C.initial_domain do
18:            H.enqueue({{v → c}, h.constraints.clone, 1.0})
19:        end for
20:    else
21:        P(h) = P(h) * C.apply(s, h)
22:        if (P(h) > τ) (for some threshold τ) then
23:            if (h.constraints = ∅) then
24:                A.add(h)
25:            else
26:                H.push(h)
27:            end if
28:        end if
29:    end if
30: end while
31: if (A = ∅ and V ≠ ∅) then
32:    return POWER(S_0, S_1.tail, C, prune(S_1.tail, V.head), H)
33: else
34:    if (A ≠ ∅ and S_0 ≠ S_1) then
35:        A = C.posit(A.head, S_0)
36:    end if
37:    return A
38: end if
```

The first time POWER is called, it constructs a priority queue of hypotheses H. Each hypothesis h is a triple $(h.constraints, h.bindings, P(h))$, where, $h.constraints$ is a set of as-yet unapplied constraints (initially S_1), $h.bindings$ is a *unique* set of candidate bindings (initially a set of mappings from the first unbound variable v in the first formula in S_1 to some set of candidate entities provided by C) and $P(h)$ is a probability value (initially 1.0) used as the priority function of H. Best-first search is then performed over the space of possible candidate hypotheses, by continuously considering the most probable hypothesis in H until H is empty. Each time a hypothesis h is considered, the following actions are performed:

1) POWER checks if all variables in the first constraint in $h.constraints$ have been assigned in h. For example, if the first element of $h.constraints$ is $near(X, Y)$, then all variables would be considered assigned if $h.bindings = \{X \rightarrow 2, Y \rightarrow 17\}$, but not if $h.bindings$ only contains $\{X \rightarrow 2\}$.
   a) If an unassigned variable v is found (e.g. Y in the example above), then h is replaced with a set of new hypotheses, each of which, in addition to the bindings already in h, contains a unique assignment to v from the set of possible candidate assignments. For example, if $C.initial\_domain = \{2, 15, 17, 18\}$, then the examined hypothesis with set of bindings $\{X \rightarrow 2\}$ may be replaced with four new hypotheses with respective sets of bindings $\{X \rightarrow 2, Y \rightarrow 2\}$, $\{X \rightarrow 2, Y \rightarrow 15\}$, $\{X \rightarrow 2, Y \rightarrow 17\}$, and $\{X \rightarrow 2, Y \rightarrow 18\}$.
   b) If no unassigned variables are found, the first constraint in S_1 not yet applied in h is applied by asking C for the degree to which the constraint applies under the candidate bindings in h, multiplying the result with h's previous likelihood, and removing the applied candidate from h's list of unapplied candidates. For example, if location 2 is very close to location 17 (e.g., with a high probability such as 0.95), the hypothesis $((near(X, Y), room(Y)), \{X \rightarrow 2, Y \rightarrow 17\}, 0.8)$ would be replaced with $((\,), \{X \rightarrow 2, Y \rightarrow 17\}, 0.76)$ (as $0.8 * 0.95 = 0.76$). If the resulting likelihood is lower than some threshold τ (e.g., 0.1), h is removed. Otherwise, if every constraint has now been applied in H, it is added to the list of candidate solutions. Otherwise it is put back into the queue.
2) Once H is empty, the set of candidate solutions A is examined. If A is nonempty or if both A and V are empty, then A is returned. Otherwise (i.e., if A is empty but V is nonempty) the resolution process is repeated with the first variable v of V removed and all constraints containing v removed from S_1.
3) If a nonempty set of solutions was returned and no constraints have been pruned away, then that set is returned. Otherwise, the best hypothesis and the original query are passed to C, which posits new entities (e.g., locations) for each variable referenced in S_0, but not appearing in the best hypothesis, and uses the formulae in S_0 to posit appropriate properties for these new entities. Finally, a new, complete solution is returned.

IV. Evaluation

To evaluate our algorithm, we used the experimental paradigm proposed in [11], which was employed to evaluate models of reference resolution under various types of uncertainty, as summarized below. While many approaches to reference resolution are targeted at a spatial domain (e.g., [1], [2], [3]), our evaluation targets the domain of descriptions of people. This domain was chosen for several reasons. First, even for this relatively simple domain, a wide variety of responses are seen; performing an evaluation on the spatial domain would cause humans results to be even more varied due to different levels of spatial reasoning ability. Similarly, the performance of POWER with respect to reference resolution would be conflated with the performance of any spatial reasoning heuristics used by its spatial reasoning consultant. Finally, while POWER is intended to be used as part of a robot architecture, we believed a full situated evaluation to be unnecessary, as previously presented domain-dependent algorithms already proven to work in robot architectures could be used as consultants for POWER, and the purview of POWER extends beyond such contexts due to its domain-independent design.

The experimental paradigm we used was designed by first considering the various types of uncertainty which may arise when resolving referential expressions. (1) In cases of *incomplete knowledge* (IK), an utterance might seem to refer to an entity not yet known to the robot. (2) In cases of *uncertain knowledge* (UK) an utterance might use properties to describe an entity which a robot is not *sure* actually has those properties. (3) In cases of *ambiguous knowledge* (AK) an utterance may seem to be equally likely to refer to multiple known entities. (4) And of course, an utterance may seem to uniquely identify an entity (a case we refer to as *certain knowledge* (CK)). These categories can apply to either the *target* of a referential expressions or to one or more of its *anchors*, i.e., the entities referenced in order to disambiguate the target. For example, in "The room at the end of the hall", "the hall" disambiguates "the room".

The experimental paradigm proposed in [11] thus defines sixteen categories of uncertainty by crossing the four categories concerning an expression's *target* (i.e., IKT, UKT, AKT, CKT) with the four categories concerning an expression's *anchors* (i.e., IKA, UKA, AKA, CKA), as well as sixteen referential expressions which each probe one of these categories within the context of a provided knowledge base. This knowledge base and set of category-expression pairs are shown, respectively, in Table I and Column 1 of Table II.

One will note that Tables I and II are each divided into three sections: in our evaluation, participants were presented with the information from the three sections of Table I separately, and were asked, for each third, questions about the corresponding third of Column 1 of Table II.

For our evaluation, 40 participants were recruited using Amazon Mechanical Turk (18 Male, 22 Female, mean age 34.75), each of whom was paid $2.00. For each referential expression, participants were instructed to specify whether each person referenced in that expression already appeared in the knowledge-base or whether they represented a new person, and to list any modifications which should be made to the list of known persons in light of the information in the examined referential expression. For example, for "Tabitha's mother", which probed condition IKT:IKA, participants were asked (1) if "Tabitha's mother" referred to anyone on the list of known persons, and if so who, (2) if the "Tabitha" in "Tabitha's mother" referred to anyone on the list of known persons, and if so who, and (3) whether any modifications or additions needed to be made to the list of known persons in light of this new informative description. The most frequent human response in each condition is shown in Column 2 of Table II. Here, referents deemed by participants not to already be in the knowledge base are denoted "?".

TABLE I: Provided Knowledge Base

ID	Name	Description
1	Jim Nelson	Doctor (**pretty sure**). Friends with Sam Greene.
2	Sam Greene	friends with Jim Nelson. **Probably** male.
3	Jim Cruz	?
4	Mary Greene	Sister of Sam Greene.
5	Frank Roberts	Jon says he's a painter, **but** Craig says he's an author **... ?** Lives next door to Nicolas.
6	Martin Francis	Painter, lives next door to Heidi.
7	Kristy Roberts	**Might** be the daughter of Frank Roberts. **Unsure**.
8	Heidi Wilkerson	Chemist, lives next door to Martin.
9	Nicolas Morris	Chemist, lives next door to Frank.
10	Craig Horton	Chemist, **might** work with Heidi? **Probably** doesn't work with Nicolas, **but who knows**.
11	Ted Wells	Baker. **Possibly** brothers with Phillip and/or Troy.
12	Phillip Wells	Brewer. **Possibly** brothers with Ted and/or Troy.
13	Troy Wells	Byron's friend. **Possibly** brothers with Phillip and/or Ted.
14	Laurie Rodgers	Byron's friend. Girlfriend of **one of** the Wells brothers.
15	Sally Owens	Teacher. Sibling of Willie Owens. Laurie's neighbor.
16	Willie Owens	Customs officer. **Possibly** female. Sibling of Sally Owens.
17	Byron Todd	**Could be** a podiatrist **... or maybe** a pediatrician.

Knowledge Base provided to participants (as originally presented in [11]). Words indicating uncertain information are presented in bold.

POWER was then provided with the same knowledge encoded in logical form, with confidence values indicative of any ambiguity attached to those statement. For example, POWER was told that Kristy was the daughter of Frank with probability 0.5. POWER was then queried with the same referential descriptions as were given to participants, encoded into logical form, with hand-annotated variable orderings.

The behavior of POWER is summarized in Column 3 of Table II. When multiple likely candidate hypotheses were found, each is listed. For these particular descriptions, whenever there are multiple candidate hypotheses, they all happen to be equally likely. Referents deemed not to be in the knowledge base are denoted "?". For those referents, POWER added new entries to the knowledge base, with prop-

TABLE II: Evaluation Results

Category and Expression	Most Frequent Human Response	POWER Responses
CKT:CKA		
The doctor's friend's sister	(Sis.:4,Fr.:2,Doc.:1)	(Sis.:4,Fr.:2,Doc.:1)
CKT:AKA		
Jim's friend	(Fr.:2, Jim:1)	(Fr.:2, Jim:1)
IKT:AKA		
Jim's daughter	(Dau.:?, Jim:1)	(Dau.:?, Jim:1) (Dau.:?, Jim:3)
IKT:IKA		
Tabitha's mother	(Mot.:?, Tab.:?)	(Mot.:?, Tab.:?)
AKT:AKA		
The chemist's neighbor	(Nei.:6, Chem.:8)	(Nei.:6, Chem.:8) (Nei.:5, Chem.:9)
IKT:UKA		
Craig's coworker's neighbor's son	(Son:?,Nei.:6, Co.:8,Craig:10)	(Son:?,Nei.:6, Co.:8,Craig:10)
CKT:IKA		
Marion's daughter Kristy	(Kri.:7,Mar.:?)	(Kri.:?,Mar.:?)
UKT:UKA		
Craig's coworker's neighbor's daughter	(Dau.:?,Nei.:6, Co..:8,Craig:10)	(Dau.:?,Nei.:6, Co.:8,Craig:10)
UKT:CKA		
Troy's girlfriend	(GF:14,Troy:13)	(GF:14,Troy:13)
AKT:CKA		
The baker's brother	(Bro.:12,Baker:11)	(Bro.:12,Baker:11) (Bro.:13,Baker:11)
AKT:IKA		
The chemist, Billie's father	(Fat.:?,Bil.:?)	(Fat.:?,Bil.:?)
UKT:IKA		
Michelle's daughter, Willie	(Wil.:16,Mic.:?)	(Wil.:?,Mic.:?)
IKT:CKA		
Sally's wife	(Wife:?,Sally:15)	(Wife:?,Sally:15)
UKT:AKA		
The Wells boy's girlfriend	(GF:14,W.B.:13)	(GF:14,WB:11) (GF:14,WB:12) (GF:14,WB:13)
CKT:UKA		
Troy Wells, the podiatrist's friend	(Troy:13,Pod.:17)	(Troy:13,Pod.:17)
AKT:UKA		
The podiatrist's friend	(Fr.:13, Pod.:17)	(Fr.:13, Pod.:17) (Fr.:14, Pod.:17)

From left to right: (1) Each condition and the referential expression used to probe that condition, (2) the most frequent human response for that condition, (3) The set of responses provided by POWER for that condition (with multiple rows used when multiple responses were returned). Cases in which the most frequent human resolution response matched one of POWER's resolution responses are bolded in Column 1.

erties and connections based on the most likely hypothesis.

V. Results

In Column 1 of Table II we depict in bold each category for which the most frequent human resolution response matched one of POWER's returned resolution responses, which occurred in 14 of the 16 conditions (87.5%). The two conditions in which POWER failed are examples of *false negatives*, in which POWER thought there was no probable match for a referenced entity. These are strictly better than the *false positives* which would have been unavoidable had the algorithm not accounted for open-world operation. False negatives are strictly better in part because they can be more easily recovered from: if it is later established that a posited hypothetical entity is in fact the same as some known, grounded entity, those two representations may be consolidated. Recovering from the discovery of an error of mistaken identity is much harder, as it would require source tracking whenever information is added to a knowledge base.

We also compare world model modifications suggested by participants with those made by POWER. Modifications made by POWER were straightforward: if POWER believed a referenced person did not yet exist in the knowledge base, it added a new representation for that person. For example, for "Jim's friend", POWER created a new representation and gave it a property indicating it was friends with Jim. In all but one condition, the most common human suggestions for world-model modification followed this pattern, and thus the most frequent human response for world model modification matched that of POWER in 13 of the 16 conditions (81.25%).

VI. Discussion

We will now examine the conditions in which POWER produced incorrect results. In condition **CKT:IKA**, POWER produces an incorrect response due to, we believe, a violation of its assumption that unknown entities are always referenced with respect to known entities. This type of violation occurs when a speaker makes incorrect assumptions about their addressee's beliefs. We believe that POWER would be able to handle this condition if it was extended to (1) consider whether newly posited anchors were highly probable matches to other known entities, (2) generate a clarification request as to whether those matches were valid, and (3) consolidate the relevant representations if an affirmative response is returned.

In condition **UKT:IKA**, participants seem to have assumed, as in condition **CKT:IKA**, some failure in belief modeling on the part of the speaker. In this condition, however, this assumption was made despite high uncertainty as to whether the *known* Willie was even of the same gender as the *described* Willie, perhaps due to the relative uniqueness of the name. In order for POWER to successfully handle this condition it would need to acknowledge that certain properties, such as being named "Willie", are relatively unique, perhaps by modeling properties' prior probabilities.

POWER's world model modifications differed from those suggested by human participants in both **UKT:IKA** and **CKT:IKA**, as would be expected. However, POWER's modifications also differed from humans in condition **UKT:UKA**, probed by the utterance "Craig's coworker's neighbor's daughter". In this condition, the response that no modification of the list was needed was more popular (by a single participant) than the response which aligned with that given by POWER (i.e., that "Craig's coworker's neighbor's daughter" or "Martin's daughter" should be added to the list). One may wonder why, for this question, the most popular human response for world model modification did not align with the most popular human resolution response. Curiously, several participants reported that "the daughter" did not already appear in the list, yet responded that no modification of the list was necessary. If these inconsistent responses are ignored, than the most popular human response aligns with the response provided by POWER. We would thus argue that for this condition, POWER provided a more appropriate response than that provided by human participants.

We would also like to discuss how we perceive POWER being used, in the context of previous approaches: it is important to recognize that POWER could be used *in conjunction* with many of the existing approaches we previously

discussed. In many cases, the core functionality of such algorithms (e.g., those presented by Williams, Fasola, Kruijf, and possibly Chai) could be wrapped into POWER consultants, possibly with an intermediate layer to mediate between known and unknown entities if the algorithm did not already operate in an open world or contain mechanisms for adding new information to its world model. POWER could then be used as the mechanism for search guiding, and the previously presented domain-specific approaches could be used for constraint evaluation and knowledge-base construction and maintenance. We believe this would be especially useful for powerful but domain-specific mechanisms such as Semantic Fields. The algorithms presented by Matuszek, Kollar and Duvallet would be less easily integrated since they parse language directly to actions and thus cannot use natural language descriptions alone to modify their world models.

Our evaluation also shows that our model is not limited to resolving references from the spatial domain, but could be used with any other domain, such as references to objects, events, or people. This is an important feature, as robots capable of natural language understanding will be expected to understand utterances which reference entities beyond simple locations. Moreover, POWER is not limited to resolving references in robotic domains; it could just as easily be applied to any intelligent agent capable of natural language interaction. However, whether POWER will scale in the domains of interest to such agents, which could have large numbers of referents, is still an open question.

Finally, we have verified the performance of POWER in the context of an integrated robot architecture where it was incorporated into the Natural Language Processing component of the DIARC architecture [12] and run on a simulated robot. POWER was able to correctly resolve referential expressions embedded in sentences such as "Jim would like the red box from the room across from the kitchen." While this was successful, we note that it would have been more advantageous if POWER had used *multiple* consultants for this type of multi-domain resolution, as we will discuss in the next section.

VII. Conclusions

In this paper, we have presented **POWER**: a domain-independent algorithm for **P**robabilistic **O**pen **W**orld **E**ntity **R**esolution which represents an important new framework in which existing technologies (i.e., previous domain-specific approaches) can be adapted to operate in an open world.

We see several important directions for extending POWER. First, it should be adapted to use multiple consultants (and thus multiple domains) simultaneously. This would allow the integration of previously disparate approaches intended to apply to different domains. This would also serve as a useful mechanism for robot architectures which store information pertaining to different domains in different architectural components (and possibly on different machines), as it would allow reference resolution to be performed without needing to consolidate information from separate domains into a single knowledge base.

POWER can also be improved by drawing upon findings from the psycholinguistics literature. For example, recent work has suggested that human natural language processing is incremental in nature, and that great performance gains can be achieved by making reference resolution incrementalized and parallelized [13]. It would be interesting to compare POWER's performance when semantic constraints are examined incremental as they are heard with performance when semantic constraints are examined according to other heuristics, e.g., constraint-ordering heuristics inspired by variable-ordering heuristics used in constraint-satisfaction problems. Similarly, it would be interesting to examine the effects of different values of the τ threshold. Finally, future work will include user studies in which the performance of POWER is tested on a physical robot.

VIII. Acknowledgments

This work was in part funded by grants N00014-11-1-0493 and N00014-14-1-0149 from the US Office of Naval Research.

References

[1] F. Duvallet, M. R. Walter, T. Howard, S. Hemachandra, J. Oh, S. Teller, N. Roy, and A. Stentz, "Inferring maps and behaviors from natural language instructions," in *ISER*, 2014.

[2] C. Matuszek, E. Herbst, L. Zettlemoyer, and D. Fox, "Learning to parse natural language commands to a robot control system," in *Proc. of the 13th Int'l Symposium on Experimental Robotics (ISER)*, 2012.

[3] T. Williams, R. Cantrell, G. Briggs, P. Schermerhorn, and M. Scheutz, "Grounding natural language references to unvisited and hypothetical locations," in *Proceedings of the 27th AAAI Conference on Artificial Intelligence*, 2013.

[4] 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 (IROS)*, 2013.

[5] G.-J. M. Kruijff, M. Janíček, and H. Zender, "Situated communication for joint activity in human-robot teams," *IEEE Intelligent Systems*, vol. 27, no. 2, 2012.

[6] J. Y. Chai, Z. Prasov, J. Blaim, and R. Jin, "Linguistic theories in efficient multimodal reference resolution: An empirical investigation," in *Proceedings of the 10th int'l conference on Intelligent user interfaces*. ACM, 2005.

[7] 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*, vol. 32, no. 4, pp. 64–76, 2011.

[8] T. Kollar, S. Tellex, D. Roy, and N. Roy, "Grounding verbs of motion in natural language commands to robots," in *Experimental Robotics*. Springer, 2014.

[9] J. Dzifcak, M. Scheutz, C. Baral, and P. Schermerhorn, "What to do and how to do it: Translating natural language directives into temporal and dynamic logic representation for goal management and action execution," in *Proceedings of the 2009 International Conference on Robotics and Automation*, Kobe, Japan, May 2009.

[10] R. Cantrell, "Mink: An incremental data-driven dependency parser with integrated conversion to semantics." in *Student Research Workshop, RANLP 2009*, 2009.

[11] T. Williams and M. Scheutz, "A domain-independent model of open-world reference resolution," in *Proceedings of the 37th annual meeting of the Cognitive Science Society*, 2015.

[12] M. Scheutz, G. Briggs, R. Cantrell, E. Krause, T. Williams, and R. Veale, "Novel mechanisms for natural human-robot interactions in the diarc architecture," in *Proceedings of AAAI Workshop on Intelligent Robotic Systems*, 2013.

[13] M. Scheutz, E. Krause, and S. Sadeghi, "An embodied real-time model of language-guided incremental visual search," in *Proceedings of the 36th annual meeting of the Cognitive Science Society*, 2014.