Agency & Information State in Situated Dialogues: Analysis & Computational Modelling

Robert J. Ross
SFB/TR8 Spatial Cognition
Universität Bremen, Germany
robertr@informatik.uni-bremen.de

John Bateman
SFB/TR8 Spatial Cognition
Universität Bremen, Germany
bateman@uni-bremen.de

Abstract

Spatially situated applications present notable challenges and unique opportunities for the dialogue modelling community. In light of this, we report on our experiences developing information-state dialogue management models for the situated domain, and present a dialogue management model that fuses information-state update theory with a light-weight rational agency model. We describe the model, report on its implementation, and comment on its application in concrete spatial language processing applications.

1 Introduction

Our work is concerned with the development of language and dialogue processing for the class of situated systems. Examples of situated systems include in-vehicle information technologies, spatially aware assistance applications, and cognitive robots. In all of these situated applications, user-system interaction through standard graphical, textual, or tactile modes of communication is either insufficient or simply not feasible for various reasons. As such, the language interface presents a highly appealing interaction mode for such applications.

Situated systems do however present notable research challenges for the dialogue community. While one noteworthy issue concerns the context-sensitive interpretation and production of spatial language that is seen frequently in the situated domain (Ross, Forthcoming), a second issue, and one which we directly address in this paper, is the *agentive* nature of situated applications. Specifically, situated applications have complex internal mental states, operate in a semi-autonomous manner, and perform actions that have clear temporal extent. Such agency features minimally require mixed-initiative and multi-threading in dialogues, but also a coupling of dialogue management with rational agency that recognizes the disparate, yet tightly coupled, nature of these elements.

We see the Information State Update (ISU) theory of dialogue management (Traum and Larsson, 2003) as being well placed to provide a basis for situated dialogue. Specifically, due to a shared lineage, ISU is a natural bridge between dialogue processes and the models of rational agency that continue to be applied within current cognitive robotics and situated systems models. But, arguably more importantly, it has now been well shown that the ISU approach is highly suited to the production of mixed-initiative and multi-threaded dialogue (Lemon et al., 2002; Larsson, 2002).

The class of classical ISU models, and in particular their realization through toolkits like TrindiKit (Traum and Larsson, 2003) and DIPPER (Bos et al., 2003) do however present some challenges when applied in the situated domain. One issue concerns the relationship between dialogue policy and the contextualization of user contributions. Within many classical ISU-based models, dialogue plans are first processed to collect mandated frame information from a user before this information is sent to a domain model for contextualization, update or query. This *collect-then-contextualize* policy favours explicit constraint gathering for complex frames, but can, if applied directly in the situated domain, lead to unnecessary clarifications and hence unnatural dialogue. To illustrate, consider the application of a *collect-then-contextualize* policy to a simple command-oriented dialogue in which the user of a robotic wheelchair attempts to direct the system to turn when the situational context makes the direction of turning clear:

(1) a. *User:* turn here
left direction is only obvious direction

b. *System:* should I turn left or right?
c. *User:* left

In such a case the clarification dialogue is superfluous and can be avoided through immediate contextualization of user contributions prior to dialogue planning policy invocation.

More significantly, due to an intended flexibility, the relationship between dialogue plans and the operations of mental state update applied within intentional systems is highly underspecified. Namely, following dialogue plan completion, domain model update information is typically flattened into a proposition set which has no epistemological form or persistence in of itself, and which must be interpreted by the domain application in an unspecified manner (Larsson, 2002). We, on the other hand, argue that scalable intelligent systems require more transparent links between the constructs of information state and the units of epistemological and intentional state.

In light of such issues, in the remainder of this paper we introduce a dialogue management model that we have developed for use in mobile robot applications. This *Agent-Oriented Dialogue Management* (AODM) model is cast within ISU theory, but (a) establishes a link between models of rationality and classical information state; and (b) applies an explicit function-based model of domain contextualization. We proceed by introducing the model's main components, followed by a description of the assumed dialogue processes, and, finally, an overview of the dialogue model's realization and application.

2 The AODM Model Components

While rejecting intractable, monolithic agent-based dialogue management models, we argue that the properties of the situated domain necessitate the inclusion of the intelligent agent metaphor in domain modelling. Thus, we apply agency models to domain organization, but capture dialogue management as meta-behaviours which operate over these cognitive constructs. In particular, we draw on techniques from the so-called *agent-oriented programing language* community (Shoham, 1993). While agent-oriented frameworks provide very rich rational agency models, here we limit ourselves to only their most salient aspects that necessarily interact with dialogue modelling and management constructs.

Taking an agent-oriented view of a domain application suggests the use of speech-act wrapped domain action and state definitions as the natural units of communication between system and user. Such a construct is essentially equivalent to a speech act in artificial agent communication languages, e.g., (FIPA, 1998). However, in natural communication, such a *dialogue move* is the result of a complex grounding process rather than a direct product of perception. Thus, following the approaches to dialogue structure originally proposed by Butler (1985) and later Poesio and Traum (1998), we assume the *dialogue act* as the primary unit of exchange at the surface/semantics interface, while assuming the *dialogue move* as the coarse grained unit of interaction established through grounding and contextualization at the semantics/pragmatics interface.

As we will see below, the *move* in classical ISU terminology corresponds more closely to our notion of dialogue act rather than dialogue move. While clearly in conflict with classical ISU terminology, our use of these terms is intended to capture two distinct levels of communicative action with meaningful terms. Moreover, this usage is derived from earlier models of Exchange Structure description used in the discourse analysis community (Berry, 1981).

In the following we flesh out these principles by detailing, first, the assumed agent components, and then the dialogue components and information state model.

2.1 Agentive Components

The main non-dialogic mental state modelling types assumed by the AODM model are briefly summarised below.

2.1.1 Capabilities

The AODM model assumes a domain agent to be endowed with one or more action definitions and zero or more plan definitions. We use the term *Capability* to generalize over actions and plans, and thus assume the agent to have a Capability Library that defines an inventory of available plans and actions. It should be noted that plan bodies can be composed dynamically outside the scope of named plan types, thus allowing a user to conjoin action and plan types arbitrarily.

We define the signatures of all capabilities, i.e., actions and plans, to have certain shared properties. First, we assume all capabilities to be performed by an agent - in our case either the dialogue agent itself or the user. Second, we assume that all capabilities have a certain earliest time at which a parametrised capability may be invoked. We may express these constructs from an ontological perspective, and assume these units to be defined in terms of the agent's conceptual ontology. Individual domains extend such signature properties into a capability hierarchy.

2.1.2 Intentions

An intention can be defined in the usual way in terms of the capability to be performed, when it is to be performed, whether there are any child or parent intentions, the state of the intention, and so forth. The intention-like corollary of a plan is an intention structure, and the agent can at any time have any number of planned or active intentions – which may be either single intentions or more complex intention structures.

The use of intentions and intention structures is of course common in both formal pragmatics and in agent-oriented applications, but for the language processing domain we minimally extend the notion of the agent's intention structure with an *Intention Salience List (ISL)*. The ISL is a stack of atomic intentions used to explicitly track the most prominent intentions within the agent's mental state. We define an atomic intention to be most salient based on recent state transitions of that intention. The ISL facilitates process resolution as required for interpreting highly elliptical process resolving commands such as "stop".

2.1.3 Beliefs & Domain State

In line with the prevalent view in the dialogue management community, we assume the details of belief state organization to be highly domain dependent. Thus, the AODM model requires only an abstract query interface over the agent's belief state. Moreover, due to the highly complex and detailed nature of spatial state, we eschew the existence of simplistic *addBelief* and similar mental state manipulation primitives in favour of specific capabilities for addressing task-specific user questions or additions of information by a user. We do however assume that unlike physical capabilities, such *cognitive* capabilities are effectively instantaneous from a user's perspective.

2.2 Dialogue Components

The AODM model also assumes a number of core dialogue components.

2.2.1 Dialogue Acts

The Dialogue Act (DA) is a conceptual-level description of a dialogue contribution made by an interlocutor. The dialogue act thus captures the semantics of individual utterances, and reflects a traditional pragmatic view of communicative function. The dialogue act may thus be informally defined as an entity which: (a) is performed by some agent; (b) potentially takes a propositional content defined in terms of the agent's domain ontology; (c) is performed at a particular time; and (d) has an associated speech function type.

2.2.2 Dialogue Moves

The Dialogue Move (DM) on the other hand is a frame-like construct that acts as the main interface between dialogue management and rational agency processes. The dialogue move is thus a more complex construct than a dialogue act – although one-to-one correspondences between dialogue acts and dialogue moves may also occur. The use of a dialogue move rather than a more complete dialogue frame was motivated by the necessity of taking an agent-oriented perspective on dialogue processing, yet building on the frame metaphor as a staging ground for meaningful unit composition.

The licensed content of a DM is directly coupled to the agent's range of capabilities and potential mental states. More specifically, user DMs and the intentions an agent may adopt are coupled in the usual way in terms of classical illocutionary logic rules which dictate that if the system is requested to perform some capability, and the system can perform that capability in the current state, then the system should adopt the intention to perform that capability.

Due to the DM's role as a construct that sits between the language interface and the agent's intentional state, we model the DM as a dynamic frame-like structure with three components:

- **The Move Template:** defines the DM type and content potential in terms of concept and role definitions extracted from the agent's conceptual ontology.
- **The Move Filler:** is the set of shallow descriptions provided by the user to fill out the

Role	Type	Filler	Solution A	Solution B
actor	Agent	nil	1.0, system	1.0, system
placement	Place	nil	1.0, here	1.0, here
earliestTime	Time	nil	1.0, now	1.0, now
direction	GenDir	GenDir modality Left	0.5, GenDir modality Left_{Ego} extent 90	0.5, GenDir modality $\text{Left}_{\text{Allo}}$ extent 90
speed	Speed	nil	1.0, normalSpeed	1.0, normalSpeed

Table 1: Instance of an `Instruct-Reorient` dialogue move for the interpretation of "turn left" in a context where both an allocentric or egocentric interpretation of "left" are possible. The first two columns define the parameter types applicable to the dialogue move in terms of concept and role restrictions. The filler column denotes the unresolved content derived from the instantiating dialogue act. The final two columns show contextualization solutions denoting alternative but equally likely interpretations of the move are denoted.

roles in the move template.

- **The Solution Set:** is the set of possible interpretations of the move filler following contextualization. While solution contents are defined in terms of the agent's application ontology, solution contents also have associated interpretation likelihoods, and typically includes content which was not directly provided by the speaker.

For illustration, Table 1 depicts a move instance which includes the Move Template, Move Filler, and Solution Set information for an instruction to make a turning, or *Reorientation*. It should be noted that for this example, the speaker provided only direction information, and that all other parameters in the presented solutions were filled through contextualization.

Though somewhat similar in nature, there are a number of notable distinctions between the DA and the DM. Unlike DAs, which can be instantiated for a broad number of speech function types, DMs may only be instantiated for task-relevant speech function types. This distinction is due to the level of non-task exchange elements being handled by the dialogue management processes without any need for explicit domain contextualization. Also, although the contents of both DAs and DMs are defined in terms of the agent's conceptual ontology, the content of a DA can be any consistent selection from this ontology, whereas the content of a DM must be headed by an application state or capability. Thus, a DM is assumed to constitute a 'meaningful' update of the agent's state rather than a fragmentary piece of information. It is then the responsibility of the dialogue process as a whole to make the mapping from fragmentary acts to complete moves.

The AODM model also applies the DM to the modelling of system initiated dialogue goals – albeit with some differences to account for the initial certainty in system rather than user dialogue moves. Essentially, unlike user dialogue moves, system dialogue moves only have a single contextualized interpretation as there is no ambiguity in system generated content.

2.2.3 Complex Components

Just as actions can be complexed into plans, and intentions into intention structures, we assume both DAs and DMs can be complexed together via semantic relations. Such modelling is necessary to capture the conjunction, disjunction, or sequencing of instructions and statements as seen frequently in situated task-oriented dialogue. We thus introduce the notion of both *Dialogue Act Complexes* and *Dialogue Move Complexes* as reified constructions of individual dialogue acts and moves. However, for the remainder of this paper we generalize the two complex sorts to their atomic constituents for the sake of brevity.

2.3 The Information State Structure

To conclude the discussion of the AODM's components, Table 2 depicts the AODM's Information State Structure. Most slot types are self explanatory, therefore we will not detail the contents of these slots here.

Slot	Type
Input Abstractions	
Latest-User-Utterance	{String,float}
Latest-User-Act	Act
User Act Containers	
Non-Integrated-User-Acts	Set(Act)
User Move Containers	
Open-User-Moves	Stack(Move)
Closed-User-Moves	Stack(Move)
System Moves Containers	
Planned-System-Moves	Stack(Move)
Raised-System-Moves	Stack(Move)
Closed-System-Moves	Stack(Move)
System Act Containers	
Planned-System-Acts	Set(Act)
Open-System-Acts	Set(Act)
Output Abstractions	
Next-System-Act	Act
Next-System-Contribution	String
Error types	
Input-Error	ErrorType

Table 2: The Information State Structure

3 Dialogue Process Models

The AODM process models and update approach follow broadly from an ISU perspective, but have been modified both due to the more action-oriented dialogues with which we deal, and to provide a more efficient implementation strategy. First, in light of the highly context-sensitive nature of situated language, we reject a strict *collect-then-contextualize* dialogue policy, and instead invoke a *contextualize-then-collect* perspective that makes use of an explicit contextualization process called immediately following the integration of user dialogue acts into the information state. This contextualization process aims to augment and resolve any resultant open user moves prior to dialogue planning. Second, to achieve a tighter coupling of dialogue and intentional behaviour, intention adoption and management strategies are integrated directly into the ISU process model. Specifically, the intention adoption strategy is integrated with dialogue planning in a single planning module, while an intention management process is invoked between response planning and the planning of concrete system messages.

Ignoring dialogue act recognition and language realization processes, the AODM control cycle can thus be summarized in terms of the following processes called in sequence:

- **Act Integration**
- **Move Contextualization**
- **Response Planning**
- **Intention Management**

Details of these processes, as well as the discourse model, are presented by Ross (Forthcoming). In the following we given a brief overview of these processes.

3.1 Act Integration

The language integration process is responsible for taking user speech acts (possibly complex) and integrating them into the information state. Successful integration of task-specific acts involves the update of open user or system dialogue moves, or the creation of new user dialogue moves. The integration process follows closely with the ISU methodology, and in particular with the general features of the model outlined by Larsson (2002) – including support for multi-threading in dialogue. As such we will not detail the model further here except to note that rather than assume a rule-based model of update, we apply in the AODM model, and in its implementation described later, a procedural approach to update specification. More specifically, while we acknowledge the importance of clear, strongly-typed, declarative models of information state, we argue that a procedural model of the update process, which is equivalent to a rule-based specification, provides a more transparent view on the update decision process, and is thus both easier to debug and extend. Moreover, we would argue that a procedural approach is in fact closer to the original view of update strategies in the *Questions-Under-Discussion* model as proposed by Ginzburg (1998).

3.2 Move Contextualization

Directly following integration, all open user moves are contextualized against the current situational model. Contextualization requires the resolution of anaphoric (in the general sense) references, elided content, and ambiguous features such as reference frame use. Due to domain complexity, we cannot view contextualization in the situated domain as simply partial unification of a dialogue move with a context model. Instead, we have developed a situated contextualization approach where functions, associated with individual semantic constituents, are used to compose concrete resolved meanings.

Rather than relying only on a set of resolution functions, i.e., functions dedicated for the contextualization of user specified content, our contextualization approach also relies on a second set of augmentation functions. Thus, each semantic role in a dialogue move type has both a resolution and augmentation function associated with it. Augmentation functions are applied in the case of a user completely omitting a move role, and typically apply default information based on situational norms – including the affordances offered by a physical context. For any given semantic role, the triggered augmentation or resolution function may produce multiple possible interpretations for that semantic role. These multiple interpretations thus result in the addition of possible solutions to a move specification as was described in Section 2.2.2. The solution set associated with a given move can both decrease and increase in size over the course of the contextualization process, and if, at the end of contextualization, more than one solution is available, the reduction of the solution set becomes the responsibility of response planning.

3.3 Response Planning

Following contextualization, the response planning process is triggered to review the information state and determine what new actions, if any, should be performed. In order to maintain synchronization between backward looking system dialogue moves and adopted intentions, the dialogue planning process is tightly coupled to the agent's intention adoption strategy. Moreover, as with language integration, we have developed our dialogue planning processes in a procedural rather than update-rule based methodology to provide greater transparency in design.

The response planning process is designed in multiple stages. The first level of response planning include the determination of what intentions and dialogue goals – if any – should be adopted. Intentions to perform requested capabilities are adopted if a requesting open user move has a single associated complete solution. Based on whether an intention is to be adopted or not, the system may also adopt an explicit dialogue move goal to signal the acceptance or rejection of particular user requests. The second level of response planning involves a lower-level choice of which dialogue acts should be assembled to pursue either new system goals or open user moves.

3.4 Intention Management

Although not a linguistic process, the AODM model also directly includes an intention management process that is responsible for sequencing adopted intentions. The justification for directly including what is usually considered a domain specific process is to ensure that sufficiently developed models of intention management, which includes the notion of the Intention Salience List as introduced earlier, are available to specific applications.

3.4.1 Illustration

To illustrate the properties of the AODM model, and in particular the relationship between units of mental state, Figure 1 depicts a dialogue example along with a partial discourse structure typical of the dialogue types that the AODM model has been designed to handle. Note that this exchange consists of two moves – the first move being a user move which requests a concrete action, and the second move being the system's response to the user move.

4 Realizing AODM with Daisie

The AODM model has been implemented within an information-state update based dialogue framework which grew out of our earlier attempts at dialogue system construction based directly on agent-oriented programming solutions. The dialogue framework, named Daisie (**D**iaspace's **A**daptive **I**nformation **S**tate **I**nteraction **E**xecutive), is a dialogue systems framework, written in Java, which provides a tightly coupled dialogue systems integration approach based on the use of a plugin architecture. An important part of our motivation in developing the system was to support a more rigorous approach to ontology definition and modularization within the description of linguistic resources and mental state. As such, the content of individual Information State slots is captured in terms of a Description Logic based representation and reasoning system.

Following earlier experiments with highly decentralized, middleware based, integration solutions, we have opted instead for a far more tightly coupled integration strategy. We argue that the future of spoken dialogue systems shall head towards ever more tight integration between ele-

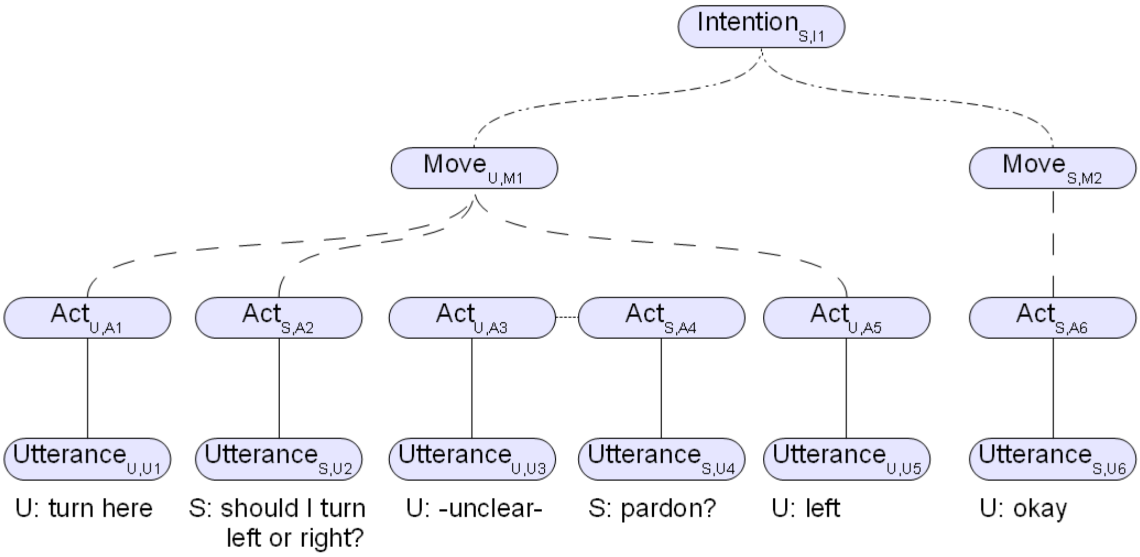

Figure 1: Inter-stratal relationships in the AODM model. Full lines denote correspondence relationships, coarse dashed lines express a constituency relationship between dialogue moves and dialogue acts, and fine dashed lines express a loose causal relationship. For each unit instance, the first subscript indicates ownership, i.e., u=user, s=system. The second subscript in turn indicates the unit instance name, e.g., u2=Utterance2.

ments where contextual information is applied at increasingly early stages to resolve ambiguity in input. While constant communication between components could be achieved through a distributed architecture, we argue that a tighter coupling between components both improves efficiency at runtime, and also improves the development process since programming interface based design rather than composing and interpreting messages is in practice easier to implement. Moreover, we argue that although a multi-agent based approach to software integration is very useful in the case of dynamic systems, typical spoken dialogue systems are very static in component design, and thus little is actually gained from a fully distributed architecture.

Ross (2008) reports on the application of an early version of the AODM model and Daisie framework to the dialogic interpretation of spatial route instructions. In this *Navspace* application, a user plays the role of a Route Giver in directing a mobile robot around a simulated office environment. The example given in Figure 1 is typical of the dialogues handled by this application. User study based evaluation of this application demonstrated that the AODM model - and in particular the contextualization process applied - led to an 86% task completion rate over 58 experimental trails conducted by 6 participants (Ross, 2008). However, this task completion rate belies the fact that participants invariable moved towards communicating their intents through very simplistic language. Integrating the AODM model with strategies that provide better context-sensitive feedback to users is thus a focus of current work.

5 Relation to Other Work

From a core modelling perspective, our treatment of dialogue moves and dialogue acts as the central representation units in a discourse representation can be considered a partial realization of Poesio & Traum Theory (PTT) (Poesio and Traum, 1998). However, whereas PTT focused on the basic tenets of the grounding process, the AODM model has been developed to explore the relationship between dialogue processes, agency and contextualization. Our consideration of the grounding process, the information state, and the problems of situated contextualization also distance the AODM model both from classical agent-based dialogue management models and also *neo-* agent-based dialogue management models such as Sadek et al. (1997)'s ARTIMIS system, or Egges et al. (2001)'s BDP dialogue agents. Within the ISU school, the AODM approach and its implementation is probably closest to Gruenstein and Lemon's Conversational Intelligence Architecture (Gruenstein, 2002; Lemon and Gruenstein, 2004). Specifically, both models advocate a tight coupling between dialogue management and agency features – although in our work we have attempted to push towards issues of representation and function-based language resolution and augmentation in an ontologically modular architecture. Finally, the function-based approach to contextualization shares motivations with recent work by Tellex and Roy (2007) in the interpretation of spatial language. However, whereas Tellex & Roy focused on the resolution of explicit language in a monologue setting, we have applied a function-based strategy to both resolution *and augmentation* in a full dialogue setting.

6 Future Work & Conclusions

We have developed and applied the AODM model to investigate the relationship between models of discourse, physical context and agency models. As such, the dialogue management model has necessarily focused on the handling of simple action-oriented dialogues. Thus, interactions typical of more complex frame structures such as booking flights cannot be handled by the current model. Instead we see frame-filling as a higher order dialogue process which operates directly on ground moves rather than un-contextualized dialogue acts. Amongst other issues, in future work we hope to investigate these relationships, and develop a frame-filling process which effectively sits above the AODM model.

Acknowledgements

We gratefully acknowledge the support of the Deutsche Forschungsgemeinschaft (DFG) through the Collaborative Research Center SFB/TR 8 Spatial Cognition – Subproject I5-[DiaSpace].

References

Margaret Berry. 1981. Towards layers of exchange structure for directive exchanges. *Network: news, views and reviews in systemic lingustics and related areas*, 2.

Johan Bos, Ewan Klein, Oliver Lemon, and Tetsushi Oka. 2003. DIPPER: Description and Formalisation of an Information-State Update Dialogue System Architecture. In *4th SIGdial Workshop on Discourse and Dialogue*.

Christopher S. Butler. 1985. Discourse systems and structures and their place within an overall systemic model. In James D. Benson and William S. Greaves, editors, *Systemic perspectives on discourse: selected theoretical papers from the 9th International Systemic Workshop*. Ablex, Norwood, NJ.

A. Egges, A. Nijholt, and R. op den Akker. 2001. Dialogs with BDP agents in virtual environments. In *Knowledge and Reasoning in Practical Dialogue Systems. Working Notes, IJCAI-2001*.

FIPA. 1998. FIPA 97 specification part 2. Technical report, Foundation for Intelligent Physical Agents, June 24.

Jonathan Ginzburg. 1998. Clarrifying utterances. In J. Hulstijn and A. Niholt, editors, *In Proceedings of the Twente Workshop on the Formal Semantics and Prgmatics of Dialogue*, pages 11–30.

Alexander Gruenstein. 2002. Conversational Interfaces: A Domain-Independent Architecture for Task-Oriented Dialogues. Master's thesis, Stanford University.

Staffan Larsson. 2002. *Issue-Based Dialogue Management*. Ph.d. dissertation, Department of Linguistics, Göteborg University, Göteborg.

Oliver Lemon and Alexander Gruenstein. 2004. Multithreaded Context for Robust Conversational Interfaces: Context-sensitive speech recognition and interpretation of corrective fragments. *ACM Transactions on Computer-Human Interaction*, 11(3):241–267, September.

Oliver Lemon, Alexander Gruenstein, and Stanley Peters. 2002. Collaborative Activities and Multitasking in Dialogue Systems. *Traitement Automatique des Langues (TAL)*, 43(2):131–154. Special issue on dialogue.

Massimo Poesio and David Traum. 1998. Towards an axiomatization of dialogue acts. In *Processings of TWENDIAL, the Twente Workshop on the Formal Semantics and Pragmatics of Dialogues*.

Robert J. Ross. 2008. Tiered models of spatial language interpretation. In *Proceedings of Spatial Cognition 08*, Freiburg, Germany.

Robert J. Ross. Forthcoming. Situated Dialogue Systems: Agency & Spatial Meaning in Task-Oriented Dialogue.

M. David Sadek, Philippe Bretier, and E. Panaget. 1997. ARTIMIS: Natural dialogue meets rational agency. In *IJCAI (2)*, pages 1030–1035.

Yoav Shoham. 1993. Agent Oriented Programming. *Artificial Intelligence*, 60:51–92.

Stefanie Tellex and Deb Roy. 2007. Grounding language in spatial routines. In *AAAI Spring Symposia on Control Mechanisms for Spatial Knowledge Processing in Cognitive / Intelligent Systems*.

David Traum and Staffan Larsson. 2003. The Information State Approach to Dialogue Management. In Ronnie Smith and Jan van Kuppevelt, editors, *Current and New Directions in Discourse and Dialogue*, pages 325–353. Kluwer Academic Publishers, Dordrecht.