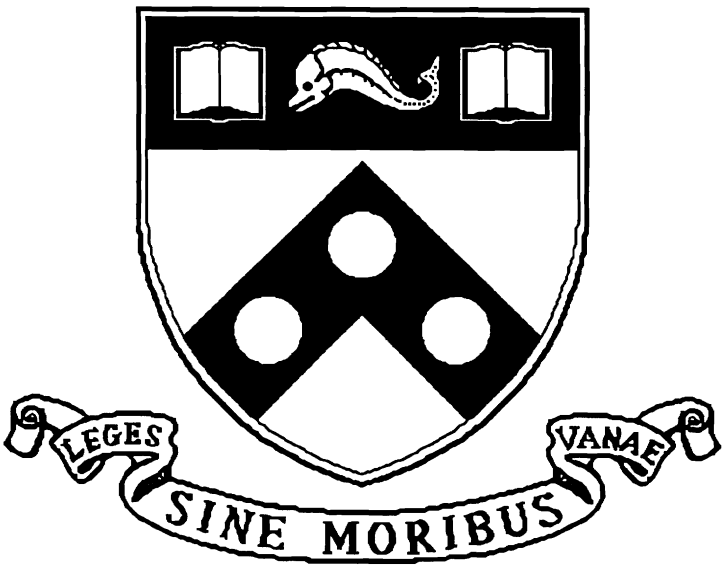
Instructions, Intentions and Expectations

MS-CIS-93-61
LINC LAB 252

Bonnie Webber
Norman Badler
Barbara Di Eugenio
Christopher Geib
Libby Levison
Michael Moore



University of Pennsylvania
School of Engineering and Applied Science
Computer and Information Science Department
Philadelphia, PA 19104-6389

June 1993
(Revised December 1993)

Instructions, Intentions and Expectations

Bonnie Webber Norman Badler
Barbara Di Eugenio Chris Geib Libby Levison Michael Moore
Department of Computer & Information Science
University of Pennsylvania*

Abstract

Based on an ongoing attempt to integrate Natural Language instructions with human figure animation, we demonstrate that agents' understanding and use of instructions can complement what they can derive from the environment in which they act. We focus on two attitudes that contribute to agents' behavior – their intentions and their expectations – and shown how Natural Language instructions contribute to such attitudes in ways that complement the environment. We also show that instructions can require more than one context of interpretation and thus that agents' understanding of instructions can evolve as their activity progresses. A significant consequence is that Natural Language understanding in the context of behavior cannot simply be treated as "front end" processing, but rather must be integrated more deeply into the processes that guide an agent's behavior and respond to its perceptions.

1 Introduction

This is a short position paper on what we have learned about language, behavior and the environment from an ongoing attempt to use Natural Language instructions to guide the task-related behavior of animated human figures. While the project, AnimNL (for "Animation and Natural Language") is not yet ready to deliver a prototype, we believe that what we have so far learned from this attempt to produce a complete vertical integration from language to animated behavior will be of interest and benefit to others as well.

AnimNL builds upon the *Jack*™ animation system developed at the University of Pennsylvania's Computer Graphics Research Laboratory. In *Jack*, animation follows from model-based

*The authors would like to thank Brett Achorn, Breck Baldwin, Welton Becket, Moon Jung, Michael White, and Xinmin Zhao, all of whom have contributed greatly to the current version of AnimNL. We would also like to thank Phil Agre, Joseph Rosenzweig, Jeffrey Siskind, Mark Steedman, Michael White, and two anonymous reviewers for their comments on the many drafts this paper has gone through. The research has been partially supported by ARO Grant DAAL03-89-C-0031 including participation by the U.S. Army Research Laboratory (Aberdeen), Natick Laboratory, and the Institute for Simulation and Training; U.S. Air Force DEPTH contract through Hughes Missile Systems F33615-91-C-0001; DMSO through the University of Iowa; National Defense Science and Engineering Graduate Fellowship in Computer Science DAAL03-92-G-0342; NSF Grant IRI91-17110, CISE Grant CDA88-22719, and Instrumentation and Laboratory Improvement Program Grant USE-9152503, and DARPA grant N00014-90-J-186.

simulation of virtual agents acting in an environment. The agents of primary interest are *Jack*'s biomechanically reasonable and anthropometrically-scaled human models (see Figure 1). The models have 138 joints, including an accurate torso, and a growing repertoire of naturalistic behaviors such as walking, stepping, looking, reaching, turning, grasping, strength-based lifting, and both obstacle and self-collision avoidance [5]. Each of these behaviors is environmentally reactive. That is, incremental computation is able to adjust an agent's performance to the situation, as the situation progresses, *without further involvement of the higher level processes* [10] unless an exceptional failure condition is signaled. Different limits can be placed on an agent's vision, strength and comfort threshold, for more realistic environmental response, and different environments can easily be constructed, so as to vary the situations in which the figures are acting.[1]

With these features, we believe that *Jack* can provide a fairly realistic target for linking Natural Language with behavior. This is because *Jack* agents "naturally" face limits on their ability to understand Natural Language utterances, much as people do: their ability to understand language relies on their knowledge, their knowledge is mediated by what they can perceive, and their perception is limited. Moreover, since *Jack* agents can, like people, effect changes on their world, their understanding of language can evolve through intentional activity in the world (cf. Section 2).

From our work with *Jack*, what we have been led to believe about language, behavior and the environment is that

- Just as an agent may be motivated by its environment to consider adopting particular goals as *intentions* that will guide it subsequent behavior, so an agent may also be motivated by Natural Language instructions. Moreover, goals recognized as being situationally relevant either through perceiving the environment or processing instructions, can clarify the meaning of other instructions. (Huffman and Laird [32] see a similar complementarity of roles between language and the environment in an agent's *acquisition* of procedural knowledge.)

- Just as an agent uses its perception of the current environment to augment its knowledge and guide its current behavior, so too can an agent use *expectations* derived from instructions about how a situation will evolve, to augment its beliefs and guide its current and future behavior.

Intentions and expectations can complement one another. Intentions can embody an agent's expectations that the agent can act in ways to satisfy those intentions, while expectations can lead an agent to form intentions to check that those expectations are satisfied and to take corrective actions if not.

[1] In discussing agents, we will use the pronoun "he", since we will be using a male figure in our illustrated example — i.e., a figure with male body proportions. The *Jack* animation system provides anthropometrically sizable female figures as well.

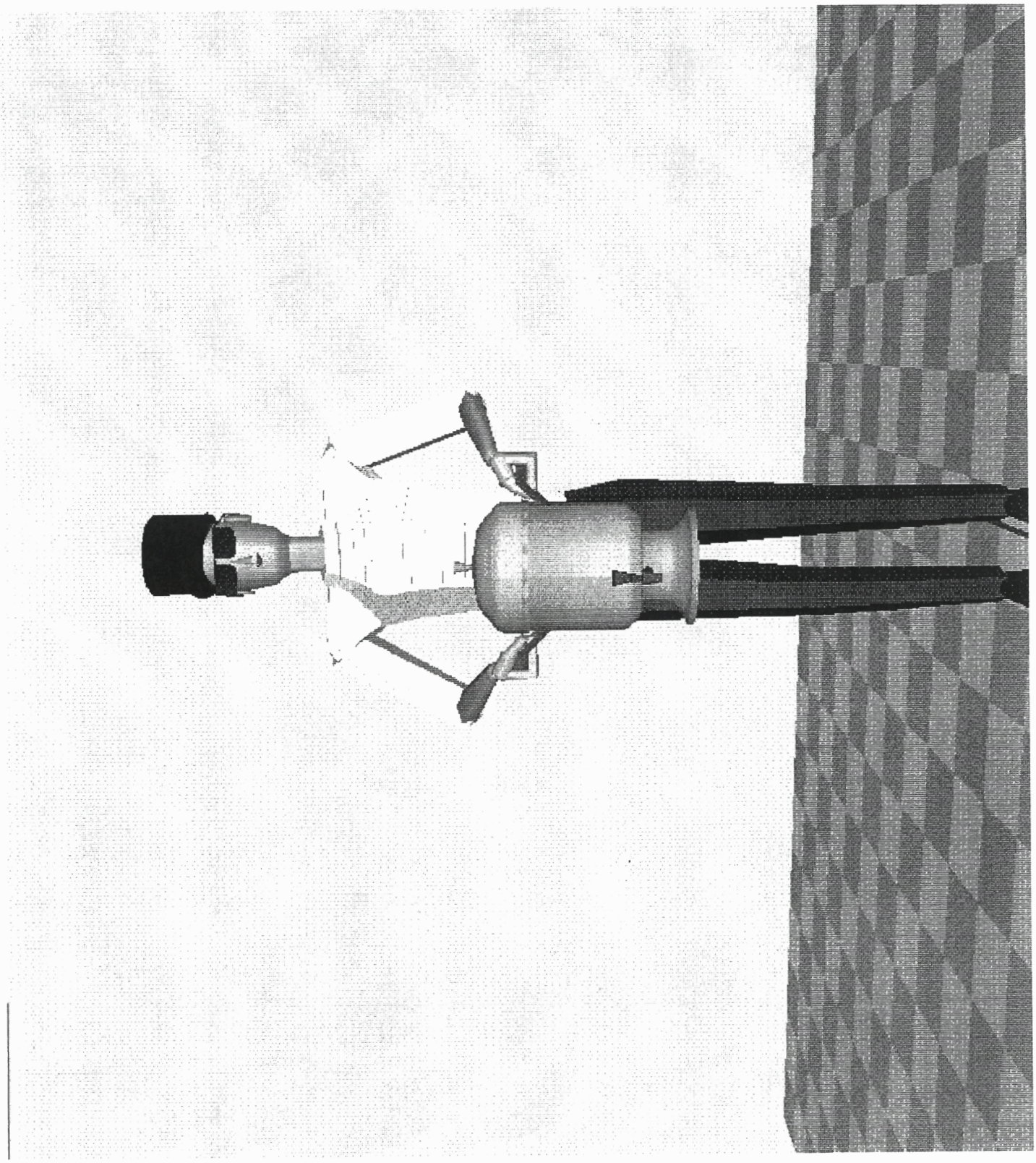


Figure 1: An Animated Agent

Three caveats are necessary here: one involving the quality of everyday instructions, a second involving the sources of an agent's intentions, and a third involving the existence of a distinct "instructor".

First, the task of formulating instructions is clearly not without problems for human language users, as the prevalence of incomprehensible and/or useless instructions shows. Two of our own favorites are:

- "Replace items on vehicle with items contained in this kit." (ANCO Replacement Windshield-Washer Pump, Stock No. 61-14)
- "To access the next highest programmed station setting, or to switch to a lower programmed station, the SCAN buttons must be repeatedly pressed." (VCR, Mitsubishi Electronics)

There are many possible things to blame for the prevalance of poor instructions, including (1) the prevalence of poor writing, in general; (2) a writer's inability to be too specific, because they lack detailed knowledge of the exact situation in which the task will be carried out; and (3) the difficulty people have in converting narrative accounts of past behavior or experience into effective instructions for future behavior. (Recall, for example, the joke about the bus rider who asks a fellow passenger where he should get off for City Hall. The latter replies, "Just get off one stop before I do".)

While significant work is being done to improve the quality of instructions [3, 25, 17, 45, 46], in the end, one may have to accept that no fixed instruction set can serve all agents in all situations.

Secondly, real agents will always have additional intentions that come from sources other than instructions: intentions arise from personal goals and desires, as well as from the policies (social, governmental, etc.) an agent agrees to adhere to. However, we have simplified the situation to one of semi-autonomous agents who have no other intentions than those that follow from their given instructions. This does not mean, however, that nothing else influences their behavior. Since *Jack* agents are environmentally reactive, features in the changing environment affect their behavior as well. For example, an agent instructed to go to the door may also have to take action to avoid obstacles it finds blocking its way. In future work, we plan to address instructions that convey general policy, and to allow it to affect agent behavior as well. For animated agent behavior that follows from particular personality traits, the reader is referred to work by Bates et al. [9, 37] and by Morawetz [14].

Thirdly, in real life there is often no distinct instructor around. In the case of multi-person tasks, this means the conduct of a task is often a product of negotiation by the participants, each with their own knowledge and beliefs [29, 47]. The intentions and expections of any one of them then may then reflect what has been negotiated by the group. We argue though that the situation with a distinct instructor can show more simply that instructions can allow an agent to form beliefs about the world that it can act on with relative confidence.

The paper is structured as follows: in the next section, we discuss plans and instructions in general, as well as some related work. We also give a brief overview of AnimNL, in support of the points we will be making about instructions, intentions and expectations. Intentions are then discussed in Section 3, and Expectations in Section 4.

2 Background

2.1 Plans and Instructions

Early views of Natural Language *instructions* corresponded to the early views of *plans*: instructions were treated as specifying nodes of a plan that, when completely expanded into primitive action specifications, would control agent behavior. This view, for example, underlies SHRDLU's successful response to instructions such as "Pick up the green pyramid and put it in the box" [54].

That *plans* should not be viewed as control structures has already been well argued by Agre and Chapman [1], Pollack [42], Suchman [48], and others in the field. Agre and Chapman show that when people form plans in response to instructions, they appear to use those plans as *resources*. The actual task situation may then lead them to interpolate additional actions not mentioned in the instructions, to replace actions specified in the instructions with other ones that seem better suited to the situation, and to ground referring terms as a *consequence* of their actions rather than as a precondition for them. Other researchers, working in what has come to be called a *BDI* framework ("beliefs, desires and intentions") now view planning in terms of agents adopting (and dropping) intentions to act [13, 16, 29, 43].

A plan's relationship with a set of instructions is also not rigid. It depends, *inter alia* on various features of the instructions, including:

- whether the instructions convey *doctrine* (general policy regarding behavior in some range of situations) or *procedure* (actions to be taken now or at some specified time in the future);
- in the case of procedural instructions, whether they are given *before*, *during*, or *after* action;
- whether the instructions are meant as *advice*, *suggestion*, *order*, *request*, *warning*, or *tutorial*.

These features are apparent in recent work involving instructed agents. For example, in Chapman's work [15], instructions are given as advice to agents already engaged in an activity. They are treated as additional evidence for an action alternative already identified by the agent as being relevant to the current situation. That alternative may not, however, be taken immediately (or ever) if other alternatives have more evidence in their favor. Chapman derives this view from observing how arcade game players follow instructions given to them by kibbitzers watching them play. (Chapman also notes that negative instructions can be similarly understood as evidence against actions.)

Vere and Bickmore treat most instructions to their "basic agent", Homer, as orders to carry out specific tasks [50]. However, they have also enabled Homer to interpret negative instructions as policy that can override orders that conflict with it. They give an example in which Homer (a small submarine) is told not to leave its island, and it subsequently refuses to comply with an instruction to take a picture of the Codfish (a ship) because to do so would require leaving the island.

Work by Alterman and his students [2] shows how instructions given after an incorrect action has been performed are treated as assistance, helping agents to accommodate their existing routines to the device currently at hand. In this approach, routines evolved over many different instances of engagement help focus an agent on the details of the situation that require attention and on the decisions that must be made. Instructions may interrupt activity to call attention to other relevant details or decisions, or to correct decisions already made. Neither plans nor instructions function as control structures that determine the agent's behavior.

Our own work has focussed on procedural instructions given to agents before undertaking a task. Such intructions can be found in user's manuals, owner's manuals, maintenance manuals, and "how to" books – for example,

- Depress door release button to open door and expose paper bag. (*Royal* CAN VAC™ Owner's Manual, p. 5.)
- Remove safety wire from access unit adjusting bolt and adjusting link and loosen bolt. (Air Force Manual T.O.IF-16C-2-94JG-50-2, Ammunition Drum, Removal and Installation)
- If candle wax falls on a piece of furniture, wait until it solidifies, then pick it off with your fingernail or a plastic spatula. (McGowan & DuBern, *Home Repair.* London: Dorling Kindersley Ltd. 1991, p. 22.)

One application that could benefit from the ability to understand and animate agent behavior that would follow from such instructions is *task analysis* in connection with Computer-Aided Design. By enabling virtual human agents to carry out maintenance and repair tasks in the CAD environment itself, a designer could determine before the artifacts were built whether people would be able to carry out those tasks in anticipated environments.

There are at least three ways to make different virtual agents interact with objects in different virtual environments. One way is through direct manipulation, which would require a designer to directly control a range of different agents in a range of different environments, in order to observe and evaluate their behavior. A second way is through direct motion sensing (e.g. "Virtual Reality" kinesthetic input [4]). The third way is through Natural Language level instructions. This could be the most economical, since it would allow the designer to simply "rerun" the same instructions with different agent-environment pairs, in order to accomplish the same ends.

Other potential application areas for using instructions to direct the behavior of animated

agents include group training activities and multi-agent simulations. Thus we feel that enabling virtual human agents to understand instructions is of practical as well as theoretical interest.

2.2 Overview of the AnimNL Architecture

We begin with a brief overview of the AnimNL architecture, since it captures our beliefs that language understanding is a process that evolves, in part, through principled interaction with the world. Roughly speaking, AnimNL consists of three interacting modules:

- A module consisting of processes that work towards understanding an instruction step in terms of an initial structure of intentions, which we call a *plan graph*. These processes include parsing, interpretation and plan inference [18, 19, 20, 21].
- A module consisting of a high-level incremental planner and two specialized processes able to adapt highly-parameterized plans for search and for object manipulation to the exact situation at hand [26, 27, 28, 36, 41].
- A simulator that coordinates motion directives and perceptual requests from the planning components with ones corresponding to environmental responses, and schedules their performance. An agenda allows multiple behaviors to be carried out in parallel, and other behaviors to be initiated and terminated asynchronously with respect to each other [10].

The result is that an agent's behavior at any time reflects both its low-level responses to the current environment and the current state of its high-level intentions. A more detailed description of the AnimNL architecture can be found in [53].

3 Intentions from Instructions

Intentions have been identified as a factor in rational behavior by various researchers (e.g., [12, 13, 15, 42]), who see them as playing at least two roles: (1) they can constrain the courses of action an agent need consider to those consistent with what the agent already plans to do; and (2) they can be used to determine when an action can be said to be relevant, to have succeeded or to have failed.

We have found that goals specified in instructions – which, in the case of positive instructions, are goals that agents are being ordered or advised or requested to adopt as their intentions – can also affect (1) how an agent interprets action descriptions in those instructions and (2) how the agent behaves in carrying out actions. In the first case, instructions perform a role that Chapman's PhD thesis [15] shows can also be performed by the environment. In Chapman's work, apparently underspecified instructions are interpreted simply as evidence for a response to the current environment that has already been deemed relevant on the basis of perception. Therefore, if a knife can be used either to kill a monster or to jimmy a door, and if a monster

is threatening the agent and no door needs to be jimmied (e.g., to enable the agent to escape), an instruction such as "use the knife" will be understood only as advice to kill the monster.

In Section 3.1, we show how goals specified in Natural Language instructions can function in a similar way, and in Section 3.2, we comment on how intentions derived from instructions can affect low-level behavioral features. We conclude in Section 3.3 with a brief discussion on how the action representation used in AnimNL allows its high-level planner to use intentions effectively.

3.1 Behavioral Import of Purpose Clauses

Comments are often made about the *efficiency* of Natural Language — how much of an utterance's meaning can be left unspecified, to be filled in by listeners able to draw on an appropriate context. In this light, we have come to understand that *purpose clauses* — infinitival clauses that convey the goal of an action – provide a helpful source of information for understanding underspecified action descriptions which convey information implicitly through context. While this information could of course be made explicit, people seem not to expect this: speakers commonly leave it for hearers to figure out for themselves.[2]

Consider the following example (from [18]):

> Place a plank between two ladders to make a simple scaffold.

The action description in the main clause is "place a plank between two ladders". The goal conveyed in the purpose clause is "make a simple scaffold". Now, by itself, the main clause conveys no explicit constraints on the orientation of what have to be two *step*-ladders and only one constraint on the placement of the plank – that it be somewhere in the 3-dimensional space "between" wherever the step-ladders are. However there are significant implicit constraints that follow from the purpose of making a simple scaffold: the ladders should be aligned with their treads facing outwards in opposite directions, at a distance spannable by the plank, which should be placed horizontally on treads of the two ladders that are the same height off the ground. (How high off the ground will depend on the purpose of the scaffold: it is not determinable from the given instruction alone.) An (incremental) plan can then be formulated to comply with both the explicit and implicit contraints on the procedure and its intended result.

Our second example is intended to show that an agent can use a goal expression and perceptual tests on when that goal is achieved to determine the referent of a noun phrase and hence what action he or she is meant to carry out. The instruction to be considered is:

> Vacuum the rug or carpet against the direction of the pile to leave it raised.

[2] There are, of course, other linguistic means of conveying purpose – free adjuncts [52], means clauses [7, 8], and even simple conjunction [24]. Moreover, clauses that convey purpose do serve other functions as well, such as making an action description easier to understand [22, 23] or justifying why an action should be done [8]. Our point here is simply that any linguistic specification of goals (purpose clauses being a clear example) can serve, like the environment, as the context in which an *underspecified* action description can be elaborated and thereby correctly understood.

To follow this instruction, an agent must know the direction referred to as "against the direction of the pile". If the agent does not know the referent of this phrase before starting, the purpose clause ("to leave it raised") can be used to guide his or her search for it. That is, the agent can plan to vacuum a bit in various directions and observe which the direction of sweep leaves the pile raised. At this point, the agent can begin to elaborate a plan for vacuuming the entire rug or carpet in that direction and thereby finish the job.

A further example is related to a matter we return to in Section 4, and concerns the termination conditions associated with perceptual tests. (The instructions are for removing wine stains from a rug or carpet.)

> Blot with clean tissues to remove any liquid still standing. Sprinkle liberally with salt to extract liquid that has soaked into the fabric. Vacuum up the salt.

In the first sentence, blotting with clean tissues specifies a *type* of activity but not the extent to which it should be pursued. (In the terminology of Moens and Steedman [40], it is simply a *process* like "running", not a *culminated process* like "running a mile": it has no intrinsic endpoint.) How long an agent should blot the stained area comes from the purpose clause "to remove any liquid still standing": the agent should plan to interleave blotting with perception, until no standing liquid is left visible.

The purpose clause in the second sentence conveys in a somewhat different way the condition under which the agent can start the final step, vacuuming up the salt. It is not the termination point of the sprinkling (which is terminated when the agent decides there is now a "liberal" amount of salt on the stain [34]), but that of the subsequent waiting. How long the agent should wait comes from the purpose clause "to extract liquid that has soaked into the fabric". The agent must plan to interleave waiting with perception, continuing until he perceives that the salt is damp (i.e., a change in visual texture). At this point, the salt has extracted as much liquid as it can, and the agent can commence vacuuming.

This example illustrates the complementary relation between intention and expectation: the intention to remove the standing liquid leads to an expectation that blotting it will eventually accomplish this removal, which in turn leads to an intention to observe the situation, monitoring for the expected point at which no liquid will be left standing.

To say the above is not to say that an agent's only intentions are those derived from instructions, but rather that goals specified in instructions, which the agent may adopt as intentions, can provide a context for fully understanding underspecified action description in instructions.

Di Eugenio has designed and implemented the machinery to be used in AnimNL for computing many of the inferences that follow from understanding that the action α described in the main clause of an utterance is being done for the purpose π described in a purpose clause. This relationship between α and π can be characterized more specifically as either *generation* or *enablement*. In *generation*, executing α under appropriate circumstances is all that is required to achieve π. In *enablement*, α brings about circumstances in which π can be generated by subsequent actions.

Di Eugenio's approach makes use of both linguistic knowledge and planning knowledge. A knowledge base of plan schemata (or *recipes*) complements a taxonomic structure of action descriptions. The latter is represented in Classic [11] and exploits *classification* to allow an inference algorithm to find related action descriptions. These descriptions index into the knowledge base of recipes which includes information about generation, enablement and sub-structure relationships between actions. The inference algorithms on these linked structures are described in detail in [18, 19].

An instruction may convey to the agent that a generation or enablement relationship holds between two actions, without the agent being able to determine which one, from the text alone. This may lead to confusion when the agent comes to act on the instruction. For example, recall the *Royal* CAN VAC instruction given in Section 2.1:

> Depress door release button to open door and expose paper bag.

(This is from a procedure for replacing the dust bag when it is full.) Whether a generation or enablement relation holds between "depressing the button" and "opening the door" will depend simply on the orientation of the canister. If the canister is horizontal when the button is depressed and the catch released, the door will fall open of its own accord because of gravity. In this case, depressing the button will generate opening the door, without the need for further action. If however, the agent has up-ended the canister to make the button more accessible, depressing the button will just release the catch: the agent must still grasp the door and pull it open. An agent who expected the former may think he didn't press the button hard enough and try again, rather than think an additional action was called for. Although this is a relatively trivial example, readers will probably recognize the problem. It could be solved by making the instruction more specific:

> Holding canister horizontally, depress door release button to open door and expose paper bag.

However, making the text longer seems to decrease the likelihood that it will be read. Trying to convey the information graphically [25] relies on the reader distinguishing between necessary features of the depicted scene and accidental ones. This suggests that producing instructions of guaranteed reliability may be an impossible task.

3.2 Intentions and Behavioral Features

In the previous section, we distinguished two relations, generation and enablement, holding between an action and its current purpose. Viewing purpose in terms of an agent's intention to achieve it, we now discuss the need we have discovered, to take account of that intention when computing low-level features of the action to be performed and animated. Such features may include the place at which the agent locates itself to perform the action, and the manner in which it grasps an object involved in the action or moves it. While our current implementation

of this capability is simply via table look-up, a more general and extensible solution is being pursued [36] for a wide class of object manipulation tasks. In the meanwhile, we believe it is still worthwhile to illustrate the phenomenon, if only by example, since such a difference can be made to human figure animation by a figure's carrying out a task-related action "naturally", even when particular behavioral choices are not necessary to simply the success of the specified action.

Our first example involves the region in which an object will be grasped. If a hammer is to be grasped simply to enable it to be moved from place to place, any region of the hammer is a viable grasp location, although somewhere near its center of mass may make it easier for the agent to lift and transport. If however, the hammer is to be grasped to enable its use in hammering a nail, a more appropriate grasp region would be towards the end of its shank. Even then, re-orientation may be required, once the hammer is lifted.

Our second example involves constraints on an agent's target in moving to a location. Consider the simple instruction

> Go over to the mirror.

By itself, this tells an agent little about where he or she is supposed to end up being positioned with respect to the mirror. On the other hand, specifying a goal that will be enabled by reaching a target location can help an agent to better identify that location. For example,

> Go over to the mirror and straighten it.
> Go over to the mirror and straighten your bow tie.[3]

Straightening a mirror requires manipulating it, so an agent will target a comfortable arm's reach. Straightening one's bow tie requires seeing oneself clearly: this may target a location either closer to or further away from the mirror, depending on the agent's eyesight. Note that the instructor may not know enough about the agent to specify a target location in more detail: it is something that only the agent can determine. Thus no further explicit guidance can be provided.

The capabilities of our existing implementation are demonstrated in an animated simulation of "SodaJack", a soda fountain agent who can respond to requests for a soda or ice cream [28]. In this domain, the intention to perform a basic task action such as moving a glass so as to *generate* serving it to a "customer", posts a constraint on the agent's movements that the glass not be tipped to one side. On the other hand, moving a glass so as to *enable* wiping it off posts no such constraint. While a few simple things such as these can be done by table look-up, the problem of systematically characterizing those features of intended actions that affect low-level physical activity is of considerable difficulty. While the significance of this boundary between symbol and action has been recognized and formalized in [33], much more work is needed in order to actually cross it.

[3] These are examples of purposive "and", mentioned earlier in footnote 2 and discussed in more detail in [24]. The "and" form sounds more natural here than the "to" form.

3.3 Intentions in Means/End Reasoning

Given that instructions only convey certain features of an agent's behavior, situated decision making is necessary to expand and amplify the agent's intentions, to fill in gaps. To enable decision making to use an agent's intentions effectively, we have found it worth replacing fixed set of preconditions, with reasoning about the effects of actions in the context in which they will be performed.

This reflects our analysis of preconditions [27] as encoding claims about the universal desirability or undesirability of certain effects of an action. For example, failing to clear off a block before picking it up, may mean that objects on top of it will slide off and break or disturb other objects. Sometimes an agent will be concerned about this possibility; other times, not. (We assume that agents may desire to avoid particular actions or actions that may lead to particular states.) The problem with fixed preconditions is that they prevent an agent from considering an action, even if the agent doesn't care about its possibly destructive side-effects.

Limited simulation, on the other hand, can enable an agent to reason roughly about the effects of performing an action in a given world state. The agent can then decide whether or not performing the action will satisfy his intentions without violating any behavioral constraints. In some cases, the consequences will be acceptable given the agent's current intentions, in other cases, they won't. This use of situated reasoning through limited simulation allows the system to use intentions to define, in a situation-specific way, when an action is applicable and can be successful rather than using an *a priori* definition. (Another use of limited simulation in planning are discussed in [38].)

The removal of preconditions from action operators has multiple effects. One benefit is to give the system more flexibility in choosing which actions to use to achieve its ends. However, the cost of this increased flexibility is that actions may fail to achieve these ends. In short, using intentions and limited simulation to perform action selection means the possibility of action failure must be allowed for.

The current version of AnimNL's planner, which eliminates preconditions in favor of situated reasoning about the effects of actions, is described in more detail in [26, 27].

4 Expectations from Instructions

Here we address the role of instructions in raising *expectations* that complement an agent's current perceptions in influencing its behavior. Expectations can lead to further perceptual activity – not just observation but also activities that enable observation. As such, expectations from instructions complement the signals coming in from the outside world. Here we discuss three types of expectations generated by different elements in instruction, and the "active perception" [6] they can engender.

4.1 Expectations about Processes

In some earlier work designed for creating animations from recipes, Karlin [34] analysed a range of temporal and frequency adverbs found in instructions. One particular construction she analysed is the following:

> Do α for <duration> or until <event>
> e.g. Steam two minutes or until mussels open.

Karlin notes that this is not a case of logical disjunction, where the agent can choose which disjunct to follow: rather, the explicit duration suggests the usual amount of time that it will take to just cook the mussels. This can be detected when all the mussels that were closed when they were put into the pot (already open ones having been discarded as dead) are now open. If they are not open after two minutes, the agent should wait a bit longer. Those that have not opened after another short wait should then be discarded, since they are full of mud.

The usefulness of an expectation such as this comes from the cost of sensing. Steaming is usually done in a closed, opaque cooking pot, so the lid must be removed in order to check the state of the contents. Whenever this is done, steam escapes, setting the process back. The result of sensing too often then is that the mussels become tough through over-cooking. The expectation can therefore be used to gauge how long to wait before beginning to make costly sensing tests.

4.2 Expectations about Consequences

Processes often have more than one possible outcome, depending on how long they proceed and how much resources they consume. Another type of expectation arising from instructions concerns the properties of objects that will result from such processes. (This is described in more detail in [51].)

Consider for example, mixing flour, butter and water.[4] Depending on the relative amounts of these three ingredients and the absorbency of the flour (different for different types of flour and for winter and summer wheat), the result may be anything from a flakey mass to a viscous batter. Instructions can indicate the intended result, so that the agent can modify and/or augment his actions so as to produce it. How instructions convey the intended result can vary: In Example 1a-c, the expected viscosity of the resulting mixture is conveyed through the verb:

> 1a. Mix the flour, butter and water, and *knead* until smooth and shiny.
> b. Mix the flour, butter and water, and *spread* over the blueberries.
> c. Mix the flour, butter and water, and *stir* until all lumps are gone.

while in Example 2a-b, it is conveyed through the noun phrase:

[4] This is not something we are capable of animating without simulating the properties of semi-viscous fluids, but it is the best example for making our point.

2a. Mix the flour, butter and water. Let *the dough* relax for 15 minutes.
b. Mix the flour, butter and water. Let *the batter* sit for 15 minutes.

There are several ways in which expectations such as these can affect an agent's behavior. The simplest is to *monitor the result*: if it doesn't meet the agent's expectation (too liquid or too solid), the agent can compensate with additional amounts of flour in the former case or water in the latter. Alternatively, the agent can *monitor the process*: that is, he can add the specified amount of water to the specified amounts of butter and flour *gradually*, mixing it in. If it is becoming too viscous, he can stop before adding all the ingredients.

4.3 Expectations about Locations

Actions can effect changes in the world that alter what an agent can perceive. Part of an agent's cognitive task in understanding instructions is therefore to determine for each referring expression, the perceptual context in which the agent is meant to find (or *ground*) its referent. Some referring expressions in an instruction may be intended to refer to objects in the currently perceivable situation, while others may be intended to refer to objects that only appear (i.e., come into existence or become perceivable) as a consequence of carrying out an action specified in the instruction.

The difference can be seen by comparing the following two instructions

3a. Go into Fred's office and get me the red file folder.
b. Go into Fred's office and refile the red file folder.

At issue is the referent of the expression "the red file folder". In (3a), it is clearly the red file folder that the listener will find in Fred's office, a file folder whose existence the listener may previously have not been aware of. That is, (3a) leads a listener to develop the expectation that *after* they perform the initial action and go into Fred's office, they will be in a context in which it makes sense to determine the referent of "the red file folder". In contrast, given instruction (3b), it is reasonable for a listener to first try to ground "the red file folder" in the context in which the instruction is given. If successful, the listener can then go into Fred's office and refile it. If unsuccessful though, a listener will not just take the instruction to be infelicitous (as they would in the case of an instruction like "Pick up the file folder", if there were currently no file folder around). Rather they will adopt the same locational expectation as in the first example, that the red file folder is in Fred's office. What is especially interesting is the *strength* of this expectation: a cooperative agent will look around, if an object isn't where they expect it to be until they find it. This has led Moore to develop flexible procedures he calls *search plans* [41] following [39], that can be used to guide an agent in grounding both definite and indefinite referring expressions. Moore's search plans are able to incorporate expectations about the context in which a referring expression will receive its intended grounding, to limit search.

In AnimNL, Di Eugenio has attempted to derive some of these expectations through plan inference techniques described in more detail in [19, 20]. In this case, the inferences are of the

form: if one goes to place μ for the purpose of doing action α, then expect to do α at μ. If α has among its *applicability conditions* – conditions that must hold for α to make sense, in terms of its potential for success in the circumstances [35, 44, 49] – that one or more of its argument be at its performance site μ, then a locational expectation develops as in (3a). If not, a weaker expectation arises, as in (3b). (Notice that this can even arise on the basis of a single clause: "Bring me the red file folder from downstairs" leads to a similar expectation as (3a), while "Give the man downstairs the red file folder" leads to a similar expectation as (3b). Haas [31], citing examples such as "Pick up the book behind you", points out a problem with indexical descriptions such as "the book behind you". A listener must decide whether such descriptions are to be grounded before they act – in this case, so that it makes sense to turn around to see the book in order to pick it up – or whether they must act on the description as given.)

In addition to expectations concerning the location of an object satisfying a particular description, an agent may also develop expectations concerning the particular description that needs to be satisfied. Here, an instruction like "Open the paint can" is more illustrative than "Get the book". An agent who simply seeks to ground the expression "the paint can" in its current situation may identify several objects of type "paint can". On the other hand, an agent who expects to be able to open the referent of "the paint can" will seek to ground a more specific expression such as "the closed paint can" or "the paint can that needs opening".[5]

5 Conclusion

The central theme of this special issue is "principled characterizations of agent-environment interactions". What we have tried to characterize in this short position paper are ways in which agents' understanding and use of instructions can complement what they can derive from the environment in which they act, lessons we have learned from attempting a complete vertical integration from Natural Language instructions to animated human figures. We have focussed on two attitudes that contribute to agents' behavior – their intentions and their expectations – and shown how Natural Language in the form of instructions provides a source of such attitudes in ways that complement the environment. We have also made the point that instructions can require more than one context of interpretation. Thus agents' understanding of instructions will evolve as their activity progresses. Understanding instructions is thus not a one-shot process that occurs entirely prior to activity. Language understanding is not just something that takes place "at the front end".

[5] This is similar, in some ways, to Haddock's "the rabbit in the hat" example [30] in which the phrase as a whole may refer uniquely in a context, even though neither of its component noun phrases ("the rabbit" and "the hat") do. Haddock's solution makes use of constraint satisfaction, the "in" relation constraining possible rabbit referents to ones that are in hats and possible hat referents to ones that contain rabbits.

References

[1] Agre, P. and Chapman, D. What are Plans for? In P. Maes (ed.), *Designing Autonomous Agents.* Cambridge MA: MIT Press, 1990, pp. 17-34. (First published as a technical report, MIT AI Laboratory, 1989.)

[2] Alterman, R., Zito-Wolf, R. and Carpenter, T. Interaction, Comprehension and Instruction Usage. *J. Learning Sciences* 1(4), 1991.

[3] E. André and T. Rist. The Design of Illustrated Documents as a Planning Task. In M. Maybury (ed.) *Intelligent Multimedia Interfaces*, AAAI-Press, 1993. (DFKI RR-92-45, 1993.)

[4] Badler, N., Hollick, M. and Granieri, G. Real-Time Control of a Virtual Human Using Minimal Sensors, *Presence* 2(1), pp. 82-86, 1993.

[5] Badler, N., Phillips, C. and Webber, B. *Simulating Humans: Computer Graphics, Animation and Control.* New York: Oxford University Press, 1993.

[6] Bajcsy, R. Active Perception. *Proceedings of the IEEE* 76(8), August 1988, pp. 996-1005.

[7] Balkanski, C. T. Actions, beliefs and intentions in rationale clauses and means clauses. In *Proceedings of AAAI-92*, San Jose, CA, 1992.

[8] Balkanski, C. T. *Actions, Beliefs and Intentions in Multi-action Utterances.* PhD thesis, Harvard University, June 1993.

[9] Joseph Bates, A. Bryan Loyall and W. Scott Reilly An Architecture for Action, Emotion, and Social Behavior. *Proc. 4th European Workshop on Modeling Autonomous Agents in a Multi-Agent World.* St Martino al Cimino, Italy. July 1992.

[10] Becket, W. and Badler, N. I. *Integrated Behavioral Agent Architecture. Proc. Workshop on Computer Generated Forces and Behavior Representation*, Orlando FL, March 1993.

[11] Brachman, R., McGuinness, D., Patel-Schneider, P., Resnick, L, and Borgida, A. Living with Classic: When and How to use a KL–ONE–like language. In J. Sowa (ed.). *Principles of Semantic Networks.* San Mateo CA: Morgan Kaufmann Publ., 1991, pp.401–457.

[12] Bratman, M. *Intentions, Plans and Practical Reason.* Cambridge: Harvard University Press, 1987.

[13] Bratman, M., Israel, D. and Pollack, M. Plans and Resource-bounded Practical Reasoning. *Computational Intelligence* 4(4), November 1988, pp. 349–355.

[14] Calvert, T. Composition of Reaslistic Animation Sequences for Multiple Human Figures. In N. Badler, B. Barsky and D. Zeltzer (eds.), *Making Them Move.* Cambridge MA: MIT Press, 1991, pp. 35-50.

[15] Chapman, D. *Vision, Instruction and Action.* Cambridge MA: MIT Press, 1991.

[16] Cohen, P.R. and Levesque, H. Persistence, intention, and commitment. In M. Georgeff and A. Lansky (eds.), *Reasoning about Actions and Plans, Proceedings of the 1986 Workshop.* Los Altos CA: Morgan Kaufmann, 1986, pp. 297–340.

[17] Delin, J. Scott, D.R. and Hartley, A. Knowledge, Intention, Rhetoric: Levels of Variation in Multilingual Instructions. *ACL-93 Workshop on Intentionality and Structure in Discourse Relations*, Columbus OH, June 1993.

[18] Di Eugenio, B. Understanding Natural Language Instructions: the Case of Purpose Clauses. *Proc. 30th Annual Conference of the Assoc. for Computational Linguistics*, Newark DL, June 1992.

[19] Di Eugenio, B. *Understanding Natural Language Instructions: a Computational Approach to Purpose Clauses.* PhD Thesis, University of Pennsylvania, August 1993.

[20] Di Eugenio, B. and Webber, B. Plan Recognition in Understanding Instructions. *Proc. 1st. Int'l Conference on Artificial Intelligence Planning Systems*, College Park MD, June 1992.

[21] Di Eugenio, B. and White, M. On the Interpretation of Natural Language Instructions. *1992 Int. Conf. on Computational Linguistics* (COLING-92), Nantes, France, July 1992.

[22] Dixon, P. The Processing of Organizational and Component Step Information in Written Directions. *Journal of Memory and Language*, 26(1):24–35, 1987.

[23] Dixon, P. The Structure of Mental Plans for Following Directions. *Journal of Experimental Psychology: Learning, Memory and Cognition*, 13(1):18–26, 1987.

[24] Doran, C. Purposive "And" Clauses in Spoken Discourse. Unpublished paper. University of Pennsylvania, July 1993.

[25] Feiner, S. and McKeown, K. Automating the Generation of Coordinated Multimedia Explanations. *IEEE Computer*, 24(10), October 1991.

[26] Geib, C. *Intentions in Means/End Planning.* Technical Report MS-CIS-92-73, Dept. Computer & Information Science, University of Pennsylvania, 1992.

[27] Geib, C. A Consequence on Incorporating Intentions in Means-End Planning. In *AAAI Spring Symposium Series: Foundations of Automatic Planning: The Classical Approach and Beyond*, Working Notes. Stanford CA, March 1993.

[28] Geib, C., Levison, L. and Moore, M. SodaJack: an architecture for agents that search for and manipulate objects. Submitted to *AAAI-94*, Seattle WA, July 1994.

[29] Grosz, B. and Sidner, C. Plans for Discourse. In P. Cohen, J. Morgan & M. Pollack, *Intentions in Communication.* Cambridge MA: MIT Press, 1990.

[30] Haddock, N. Computational Models of Incremental Semantic Interpretation. *Language and Cognitive Processes* 4, 1989, pp. 337-368.

[31] Haas, A. Natural Language and Robot Planning. Technical Report, Dept of Computer Science, SUNY Albany, September 1993. Submitted to *Computational Intelligence.*

[32] Huffman, S. and Laird, J. Acquiring Procedural Knowledge through Tutorial Instruction. *Proc. 1994 Workshop on Knowledge Acquisition for Knowledge-Based Systems.* Banff, Canada, January 1994.

[33] Israel, D., Perry, J. and Tutiya, S. Executions, Motivations and Accomplishments. *The Philosophical Review*, forthcoming.

[34] Karlin, R. Defining the Semantics of Verbal Modifiers in the Domain of Cooking Tasks. *Proc. 26th Annual Meeting, Association for Computational Linguistics*, SUNY Buffalo, June 1988, pp. 61-67.

[35] Litman, D. and Allen, J. Discourse Processing and Commonsense Plans. In P. Cohen, J. Morgan & M. Pollack, *Intentions in Communication.* Cambridge MA: MIT Press, 1990, pp. 365–388.

[36] Levison, L. How Animated Agents Perform Tasks: Connecting Planning and Motor Control Through Object-Specific Reasoning. *AAAI Spring Symposium on Physical Interaction and Manipulation*, Stanford CA, March 1994

[37] A. Bryan Loyall and Joseph Bates. Real-time Control of Animated Broad Agents. *Proc. 15th Annual Conference of the Cognitive Science Society* Boulder CO, June 1993, pp.664-669.

[38] McDermott, D. Transformational Planning of Reactive Behavior. Research Report 941, Computer Science Department, Yale University, December 1992.

[39] Miller, G., Galanter, E. and Pribram, K. *The Structure of Plans and Behavior.* Holt, Rinehart and Winston, Inc. 1960.

[40] Moens, M. and Steedman, M. Temporal Ontology and Temporal Reference. *Computational Linguistics*, 14(2):15-28, 1988.

[41] Moore, M.B. *Search Plans.* PhD Dissertation proposal, Department of Computer and Information Science, University of Pennsylvania, May 1993. (Technical report MS-CIS-93-55).

[42] Pollack, M. *Inferring domain plans in question-answering.* PhD thesis, Department of Computer & Information Science, Technical Report MS-CIS-86-40. University of Pennsylvania, 1986.

[43] Rao, A. and Georgeff, M. An Abstract Architecture for Rational Agents. *Proc. KR-92*, Boston MA, 1992, pp.439-448.

[44] Schoppers, M. Universal Plans of Reactive Robots in Unpredictable Environments. *Proc. Intl. J. Conf. on Artificial Intelligence*, Milan, Italy, 1987.

[45] Schriver, K. Plain Language through Protocol-aided Revision. In E. R. Steinberg (ed.), *Plain Language: Principles and Practice.* Detroit MI: Wayne State University Press, 1991, pp. 148-172.

[46] Schriver, K. Teaching Writers to Anticipate Readers Needs: A Classroom-Evaluated Pedagogy. *Written Communication*, 9(2), 1992, 179-208.

[47] Sidner, C. Using Discourse to Negotiate in Collaborative Activity: An Artificial Language. *Proc. AAAI Workshop on Cooperation among Heterogeneous Agents*, San Jose CA, July 1992.

[48] Suchman, L. *Plans and Situated Actions.* New York: Cambridge University Press, 1987.

[49] Tate, A. Generating Project Networks. *Proc. Intl. J. Conf. on Artificial Intelligence*, Milan, Italy, 1987.

[50] Vere, S. and Bickmore, T. A basic agent. *Computational Intelligence*, 6(1), 1990, pp.41-60.

[51] Webber, B. and Baldwin, B. Accommodating Context Change. *Proc. 30th Annual Conference of the Assoc. for Computational Linguistics*, Newark DL, June 1992.

[52] Webber, B. and Di Eugenio, B. Free Adjuncts in Natural Language Instructions. *Proc. COLING*, Helsinki Finland, July 1990, pp. 395-400.

[53] Webber, B., Badler, N., Baldwin, F., Becket, W., Di Eugenio, B., Geib, C., Jung, M., Levison, L., Moore, M. and White, M. "Doing What You're Told: Following task instructions in changing but hospitable environments". In Y. Wilks and N. Okada (eds.), *Language and Vision across the Pacific*, to appear 1993. (Also appears as Technical Report MS-CIS-92-74, Department of Computer & Information Science, University of Pennsylvania.)

[54] Winograd, T. *Understanding Natural Language*. New York: Academic Press, 1972.