Coördinating Human-Robot Communication

by

Andrew G. Brooks

B.Sc.(Hons), University of Adelaide, 1994
Dip.Ed., University of Adelaide, 1995
M.Sc., Australian National University, 2000

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in the field of Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2007

Author .....
Department of Electrical Engineering and Computer Science
November 30, 2006

Certified by .....
Cynthia L. Breazeal
Associate Professor of Media Arts and Sciences
Thesis Supervisor

Accepted by .....
Arthur C. Smith
Chairman, Department Committee on Graduate Students

Coördinating Human-Robot Communication

by

Andrew G. Brooks

Submitted to the Department of Electrical Engineering and Computer Science
on November 30, 2006, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in the field of Electrical Engineering and Computer Science

Abstract

As robots begin to emerge from the cloisters of industrial and military applications and enter the realms of coöperative partners for people, one of the most important facets of human-robot interaction (HRI) will be communication. This can not merely be summarized in terms of the ongoing development into unimodal communication mechanisms such as speech interfaces, which can apply to any technology. Robots will be able to communicate in physically copresent, "face-to-face" interactions across more concurrent modalities than any previous technology. Like many other technologies, these robots will change the way people work and live, yet we must strive to adapt robots to humans, rather than the reverse. This thesis therefore contributes mechanisms for facilitating and influencing human-robot communication, with an explicit focus on the most salient aspect that differentiates robots from other technologies: their bodies.

In order to communicate effectively with humans, robots require supportive infrastructure beyond the communications capabilities themselves, much as do the humans themselves. They need to be able to achieve basic common ground with their counterparts in order to ensure that accurate and efficient communication can occur at all. For certain types of higher level communication, such as skill transfer, robots need some of the underlying cognitive mechanisms that humans both possess and assume to be present in other communicative agents. One of these general mechanisms is self-awareness. This thesis details development of these underlying infrastructure components.

Four broad areas of human-robot communication are then investigated, and applied to four robotic systems with different physical attributes and computational architectures. The concept of *minimal communication*, in which a robot must communicate basic information without the benefit of immediately recognizable anthropomorphic features, is presented. A system for enabling *spatial communication*, in which the human and robot must achieve common ground and support natural physical communication in the presence of other physical objects in the shared environment, is described. A model for behavioral encoding of *non-verbal communication* is developed, including the expression of both body language and proxemics. Finally, the use of existing communications modalities to produce *interactively shaped communication* for future expression is introduced, through a system that allows a human director to coach a robot through an acting performance.

The robots featured in this thesis are the "Public Anemone" interactive robot theatre exhibit and "Leonardo" humanoid robot interaction testbed of the MIT Media Laboratory's Robotic Life Group; the "Robonaut" autonomous humanoid astronaut assistant robot of NASA Johnson Space Center's Dextrous Robotics Laboratory; and the "QRIO" autonomous humanoid entertainment robots of Sony Corporation's Intelligence Dynamics Laboratories.

Thesis Supervisor: Cynthia L. Breazeal
Title: Associate Professor of Media Arts and Sciences

Acknowledgements

Such a long, twisted path stretches back to the horizon. How did I get here from there?

First and foremost, I owe my greatest debt of gratitude to my parents, Rosemary Brooks and Simon Stevens, for their inspiration and unwavering support. If I have to grow up, I hope it's like them.

I remain grateful to everyone who helped me come to the Institute — those who wrote recommendations, such as Peter Eklund, Mike Brooks and Alex Zelinsky; Bill Townsend, who gave sage advice; and especially Tom von Wiegand, who shepherded my application into RLE, and has been an indispensable mentor and friend ever since.

Particular thanks, of course, are due to my advisor at the Media Lab, Cynthia Breazeal, for taking a chance on, and continuing to believe in — some might say put up with — me, and for making the Robotic Life group such a unique and wonderful place to be.

To my fellow denizens of Robotic Life — Jesse Gray, Matt Berlin, Guy Hoffman, Andrea Lockerd Thomaz, Dan Stiehl, Cory Kidd, and Jeff Lieberman — I couldn't have done it without you. It may sound clichéd, but it's true.

Hails to:

David Demirdjian, for generously letting me use his code to get started doing vision on the PC.

Richard Landon and everyone at Stan Winston Studio, for making sure Leo was there when I needed him.

Bill Bluethmann and the folks at the NASA JSC Dextrous Robotics Lab, for welcoming me into the halls of the Robonaut.

Ron Arkin of Georgia Tech, and Hideki Shimomura, Yukiko Hoshino, Kuniaki Noda, Kazumi Aoyama, and all the rest of the SIDL staff, for giving me a chance to play with QRIO.

My readers, Leslie Kaelbling and Trevor Darrell, whose comments improved this thesis.

Dr Michael Wong Chang and Dr Shlomo Abdullah, for their beautiful solid modeling, and to all the other good people and things that distracted me along the way.

Contents

Chapter 1

Introduction

Think about a society in which robots live and work with humans. These humans may live much as industrialized humans do now; the robots perform a variety of roles such as assistance, protection and entertainment. This is the direction in which much current robotics research would lead us. However, such robots should not be merely personal computers with arms. In order for them to progress from simple tools to coöperative partners, capable of both learning from people and teaching them, we will need to be able to communicate with them as we do with other people and living things, not as we do with more passive technologies. The purpose of this thesis research is to improve the design of robots so that this need can begin to be satisfied.

This work thus fulls under the general umbrella of human-robot interaction, customarily abbreviated HRI. This field is still a young one — itself often considered a specialized offshoot of the wider area of human-computer interaction (HCI), it is still lacking in domain-specific terminology and definitions (e.g. [91]). Numerous promising results indicate that HRI researchers can validly draw upon lessons from both the HCI literature and the broad body of work on human-human social communication (e.g. [37, 54, 150, 297]), but a large amount of basic experimental work is yet to be done concerning the special characteristics of interactions with robots.

Since interactive robots are still such a new and rare technology, such basic experimentation is prone to contamination from novelty effects and unrealistic expectations from science fiction. Thus, we must build more complex, communicative robots, and more of them. Informed by the existing literature, this process essentially involves interface design and the management of expectations — taking into account the natural communication strengths of humans, and solid assumptions about what differentiates robots from other technologies.

One such clear difference is the way in which the robot is physically situated — the fact that it has a body that can react to and affect the physical environment in causally perceptible ways. All of the communication schemes designed in this thesis incorporate some facet of bodily communication. In particular, they attempt to improve the human's understanding of the robot through its use of its body, and the robot's understanding of the context in which its body participates in the communication.

At this point, it is appropriate to define the terms used in the title of this thesis. "Coördinating" refers to the aforementioned interface design process — the integration of multiple modalities into an effective and robust interactive mechanism. The term "Human-Robot" is used to specifically

imply a face-to-face, copresent interaction between one or more humans and the robot. "Communication" refers to the process of sharing information between the interaction participants, a process which requires the establishment and maintenance of common ground according to the prevailing expectations, and the ability to estimate and improve the quantity and relevance of the knowledge that is being shared.

The methodology used in this thesis centers on the application of design principles and ideas to 'live' interactions with real robots. I have attempted to select and integrate currently available communications technologies according to theories from a diverse range of literature domains in order to produce robust multimodal demonstrations and novel design models. Achieving these outcomes has involved a process of designing to the inherent strengths of the human and the robot, of keeping the human 'in the loop' where it is beneficial to do so, and of embracing modularity by focusing on the larger algorithmic architecture rather than the limitations of the underlying components.

1.1 Motivations

The motivation for this research into designing robots that are better at communicating is primarily to enable a practical distinction to be drawn between robot users and robot designers. In stating that the goal is to create robots that humans can efficiently and effectively communicate with and understand, the directive is implicitly inclusive — it means *all* humans, not just those possessing prior expertise in robotics. Many application areas flow from this precept: from expansion into technology-averse populations such as the elderly, to customization and personal instruction of the robot by users with diverse existing skillsets.

In order for non-roboticists to be able to communicate expertly with robots while remaining non-roboticists, it is clearly necessary for the robot to communicate using modalities and mechanisms that humans largely find already familiar and intuitive. Indeed, many of the current motivating trends in HRI concern aspects of 'natural' interaction and concerns for facilitating familiar styles of human interaction. This work crosses paths with many of these trends.

One of the more formal trends in HRI is to continue the HCI focus on strict metrics for interface design, measured in terms of features such as situational awareness, workload, efficiency, etc. (e.g. [2, 55, 71, 91, 98, 206, 265]). This avenue of research proceeds according to theories stating that natural, intuitive and/or familiar interfaces to generally require less operator training and are less prone to user error or confusion. Thus the motivations of this thesis are aligned with this trend, because enabling non-expert users is equivalent to requiring users to possess a smaller quantity of specialized training and experience in order to use the robot effectively.

Similarly, a related popular motivation of human interactive robotics is to improve human task performance by creating assistive robots capable of working in human environments (e.g. [27, 144, 225, 284]). This work seeks to minimize the environmental adaptation humans must accept in order to take advantage of a robot collaborator. This thesis work shares this motivation, because the means of communication are features of the task environment. In addition to championing natural human modes of communication, an infrastructural motivation underlying the work in this thesis has been to make use of non-intrusive sensing methods, such as untethered visual and auditory processing, wherever practical.

An increasing number of researchers into autonomous and socially intelligent robots have been investigating human-robot teaching and general methods for robot learning from demonstration (e.g. [43, 46, 214, 261]). Being able to update a robot with new skills, or further refine old ones, or even to simply customize the robot with one's personal preferences, would be a boon from the point of view of both the robot designer and the end user. This thesis work therefore shares this general motivation; but more importantly, as an advocate of retaining strong human influence on the robot design process as well as the robot's eventual activities, I recognize that learning from human instruction will be essential to fill in the unavoidable gaps in our ability to exhaustively design robots for particular contingencies. Enabling demonstrative communication is therefore a strong motivation within this work.

An emerging trend in research into robotic architectures and applications is examining the entertainment abilities of robots (e.g. [42, 52, 50, 101, 300]). Motivations here are partly commercial, of course, but they are also related to the motivation of having robots learn from humans. This is simply because even if a robot is capable of being taught by a human, it will acquire no benefit from this ability if humans are not motivated to teach it. The teaching process must be entertaining in order for this to occur. For this to be true — and also to enable display entertainment applications such as acting — the robot must be fluid and natural, with accurate use of nuance, subtlety and style in order to hold the interest. This thesis is particularly motivated by the need to incorporate these qualities in human-robot interactions.

Another high-profile application area of human-robot interaction research is the field of therapy and palliative care, particularly eldercare (e.g. [165, 239, 249, 295]). This is an area in which sensitivity to the target clientele is paramount, especially in terms of providing a convincingly genuine interaction. Robust common ground is also essential, and co-motivation through team building can be important in clinical environments such as rehabilitation. This thesis work is also motivated by the goal of providing guaranteed baseline interactivity and improving common ground, as well as working towards firm human-robot partnerships.

Of course, a perennial trend in robotics is hardware design, and the area of this work most relevant to this thesis is research into the design of more expressive robots, most of which focus on more accurate anthropomorphism (e.g. [99, 110, 129, 173, 205, 312]). While this thesis work is not directly motivated by a desire to contribute to hardware design directly, it does make use of robots designed with advances in expressive hardware and aims to examine how to make good use of that hardware for communicative purposes. In turn, this may help to isolate areas most worthy of future concentration.

Finally, a small but crucial trend in human interactive robotics and related technology is the basic experimental psychological research that seeks to tease out the true parameters affecting human reactions to interaction designs, such as expectations and social compacts, through human subject studies (e.g. [45, 54, 170, 204, 240, 245, 297]). This thesis, as essentially an engineering document, does not attempt to make such basic research claims — user studies in heavily engineering-based HRI projects notoriously attract criticism for generating simplistic results that merely justify the project according to some cherry-picked metric, and for being unrepeatable due to the unavailability of identical robotic hardware — but it is strongly motivated by such research to inform its design choices, and to set the stage for such studies where appropriate. For example, a carefully designed communication scheme can utilize HRI studies to verify the suitability of the robot itself, whether or not said scheme operates autonomously or is achieved via a Wizard-of-Oz method.

1.2 Problems Addressed

The primary goal of this thesis is to explore improved ways in which robots can make good use of the defining characteristic of their technology — their bodies — to communicate and maintain common ground with humans, with a particular emphasis on enabling fruitful interaction with those humans who are not expected to have design knowledge of the robot's architecture. The intention is not, however, to claim that any detail is indispensable or that any implementation is definitively "the way it should be done". Communication is a complex discipline, humans are expert at making it work in a variety of different ways, and interface designers frequently disagree.

Instead, this thesis presents designs that demonstrate increased possibilities, and supporting arguments for the potential merits of approaching the relevant problems in these ways. All robots are different, as are the circumstances in which they will interact with humans — from nursing homes to outer space — and I believe that creative human design and careful attention to the problem domain will trump rigid or "one size fits all" methodologies for the foreseeable future. With that said, these are the questions within human-robot communication that I have sought to address with the designs in this thesis.

How can a robot understand humans?

Or, how can humans adapt the robot's understanding of communicative partners in general to themselves in particular?

Humans share certain aspects with all other humans. When communicating, they can assume a range of things and accurately predict responses and shared knowledge about what it means to be human. Robots are equipped automatically with none of these things, and we don't yet (and may never) know how to design all of these things in. We need ways for human users of robots to be able to make sure the robots understand certain basic qualities about them — in essence, to be able to calibrate the robot to a human or humans — and these methods need to be sufficiently brief, simple and easy that robot users will actually be motivated to do them.

One specific example is the ability of a robot to map the human's body onto its own, in order to understand the human's physical situation in the context of its own self-understanding. Humans exhibit many individual differences that are immediately apparent to other humans, but are unlikely to be discernible by robots given current sensing technology and contextual models. I have addressed this with an interaction designed to teach the mapping quickly and enjoyably using visual sensing.

How can a robot understand itself?

Or, how can the robot accurately model its own body and connect that model to physical consequences and contextual references?

Humans learn their own body contexts through years of experiencing them through their specialized neural hardware. It is not realistic to expect robots to do this in precisely the same way, nor humans to intentionally design such a time-consuming process. We need a way to bootstrap this outcome based on design insights. Experiential learning can then be used to fill gaps that are difficult to resolve with design.

More specifically, I argue that the robot needs an internal simulator with forward models that allows it to estimate action results without performing real-world trials, and to test conversions between internal representations (such as stored motions). When communicating with external entities such as humans, it also need a way to physically ground certain communications that concern its own body. I have addressed these problems with an internal self-imagination that reuses all of the robot's original cognitive structures such that it does not require additional mechanisms to interpret the results, and with an internal attention mechanism that is fed by the robot's real and imagined behavior in order to disambiguate body-related communications.

How to ensure communication is possible?

Or, how can we ensure the capabilities exist for interactive human communication to occur when they may conflict with the physical design of the robot?

Humans tend to have, by and large, fairly similar design. Yet even among humans, in cases where this is not true, conflicts can occur, even over aspects as trivial as race or appearance. Humans are also experts at using human concepts to interpret non-human behavior; creatures that are better (or simply more fortunate) at exploiting this tend to receive certain advantages in their dealings with humans (e.g. dogs). Yet robots can be designed that are at best on the fringes of human physical intuition, yet still must communicate critical information. Robot designers need a methodology for ensuring this can occur.

In this thesis I submit that some sort of baseline performance is needed. However it is not yet certain what such a baseline performance might require. How does an unfamiliar robot communicate convincingly? What is most important to know in order to do this, if the environmental constraints prevent optimal sensing? I discuss these issues in terms of concepts from the literature on human-computer interaction, suggest definitions for appropriate baseline conditions, and present a project that required adherence to these strictures.

How to communicate about the world?

Or, how can a human communicate with a robot about physical items in the world when the robot tends to know so much less about them?

Humans recognize objects and settings largely by their context, inferred once again from their long term experience, but we are far from being able to design all of this contextual knowledge into a robot. However, communication can be used to help the robot record details about objects conforming to the simple representations of which it is currently capable. We need to design efficient methods of facilitating human explanation involving objects, so that this process is feasible in real situations.

Specific aspects of this problem include ways to refer to objects in complex sensing environments with restricted communications modalities, such as outer space, and ways to support typical human communications behavior, which can involve imperfectly predictable variation among modes of reference, the tendency towards minimal effort through reliance on built-up context, frequent perspective shifts, and so on. I have addressed this problem by developing comprehensive support for a particular form of object reference — deictic reference — involving a "point to label" paradigm

which seeks to build up specific contextual labeling to allow subsequent modality reduction without ambiguity.

How can a robot express itself physically?

Or, how can the robot communicate certain internal information using the same subtle non-verbal cues that humans are trained to recognize?

Humans recognize certain details of the internal states of other humans from physical cues that manifest themselves externally — often unconsciously — as a result of these states. These manifestations tend to exhibit cultural dependence, so once again humans refine this capability through long-term experience. Robots do not, of course, automatically manifest these cues, unless they have been designed to do so. Rather than expect humans to learn new sets of subtle cues for each type of robot, it is likely to be more efficient to have robots try to express the cues humans already know. Is this possible to do convincingly, and if so how can we effectively design such communication?

More specifically, since the human capability for subtle physical communication is so broad, how can we implement a range of non-verbal cues on a robot, particularly if it is not equipped with the capability to form caricatured facial expressions? In particular, how can we implement proxemic communication, which is affected not only by state information such as emotion, but is also dependent on metadata concerning the state of the relationship between the communicative parties and the proxemic context of the interaction itself? I address these questions by designing non-verbal communication for a humanoid robot containing a suite of possible kinesic and proxemic outputs, and data structures to represent the information needed to support recognizable proxemic behavior.

How best to design for the future?

Or, how can we design human-robot communication schemes to easily take advantage of inevitable advances in the underlying technology?

Research into better algorithms for single communications modalities is essential, but integration of these techniques can cause major headaches when systems have been designed according to assumptions that new technologies render obsolete. Software engineering methodology teaches us to embrace modularity, but this requires standard interfaces between modules. What the standard should be for cognitive components such as perception or memory is an open question, but robot designers should still attempt to coördinate their designs so that improvements such as more comprehensive perceptual systems, better models of world awareness, or expanded linguistic vocabularies, can be integrated without major cognitive replumbing. In many of the designs in this thesis, I have attempted to address this issue by setting up basic, efficient component interfaces, and have been able to successfully substitute updated components on a number of occasions that are mentioned in the text.

This question is specifically relevant to physical communication, because if a robot uses its body to communicate as well as to do tasks, we need to try to design its architecture so that we can improve the communication when possible without needing to rewrite all of the tasks that occur

concurrently — and vice versa. In other words, to use modularity to partition behavior design (in the behavioral robotics sense) from communications design. I have addressed this by looking at ways to encode bodily communication by injecting it into the stream of physical output of existing behaviors, rather than generating it separately.

How can particular communications be taught?

Or, how can a person interactively construct complex communications expressions for robots without needing to know how the robot works?

Methods for generating specific, detailed expressive sequences from a robot without access to the robot's software architecture are rare, and assembling long sequences of actions by prespecifying them without ongoing feedback is error prone. Many aspects affect the overall impression created by a communication sequence, and it can be difficult to predict in advance how a component-based sequence will actually appear to an observer. An interactive approach could be ideal, but once again the robot also cannot yet know about or understand all of these aspects, nor even robustly recognize the general contexts in which they are appropriate. How can interactivity be used to accomplish this in the presence of the disparity in context awareness between the human and robot? This raises many further underlying questions.

In particular, if a robot's actions are triggered by beliefs caused by its perceptions, how can this architecture support sequencing of actions according to subtle timing for communicative effect? How can the human shape a communication sequence once it has been taught, and how can he or she communicate to the robot which particular part of the sequence is to be altered? If a robot has more than one type of internal motion representation, how can a human teacher adjust the style of a movement without needing to know the format in which it is stored, and how can the human and the robot agree on the meaning of a particular style term in the first place? How can the robot communicate its understanding of any of this in return?

I address all of these questions in this thesis. I describe a system for enabling the human to generate arbitrary expressive content on the robot by assembling atomic actions that the robot already knows into multimodal timed sequences that can then be adjusted and rearranged in time. Furthermore, the styles of the actions can also be modified and overlaid, in a transparent fashion that requires the teacher to have no preëxisting knowledge of the robot's motion architecture, nor the internal stylistic reference symbols specified by the robot's designer. The system is designed as a real-time tutelage interaction in which the robot constantly communicates its understanding back to the human to maintain shared awareness of the interaction.

1.3 The Robots

During this thesis work I have had the great fortune to be able to apply my ideas to several robots, each of which is among the world's most advanced in its respective application domain. As one's own personality is partly shaped by the personalities of those people with whom one finds interaction inspiring, so too this thesis has been shaped by the unique architectures and design challenges presented by each of these robots. Though they do not have personalities as people do, their idiosyncrasies define them in curiously similar ways. I would therefore like to introduce them

individually as one would old friends, summarizing their origins and the course of events which led me to be involved with them.

Notwithstanding their individual differences, they all shared the essential quality of possessing a body, and they all needed to use it to communicate with humans, so I will also specify what was physically special about them in this respect.

Public Anemone (Figure 1-1) was the first robot that I worked on upon coming to the Media Lab. As a project, it fulfilled a number of concurrent roles. Within our group, it served as a skill synthesizer, allowing each of us to apply the technologies we had been working on to a concrete realization; and as a team bonding experience, as we sweated away together to give birth to it in a Medford warehouse in the summer of 2002. Without, it served as a technology demonstrator and expressive robotics proselytism, spreading our ideas to visitors to its exhibit in the Emerging Technologies section of SIGGRAPH 2002.

Figure 1-1: Public Anemone, an organic robot actor.

All of us started work on Public Anemone based on what they were doing at the time, but as the project figuratively took on a life of its own, many of us found ourselves being led in new directions. I had started with computer vision, but as the creature design work progressed in increasingly weird ways (the original plan called for the entire robot to be submerged in a tank of mineral oil), I became ensnared in thoughts of what it meant for such a robot to be truly interactive, how it could be convincing and natural under baseline conditions, how humans would ultimately try and engage with it, and what its appropriate responses to that engagement should be.

As the above implies, the sheer physical unusualness of the Public Anemone was what made it unique. With its alien, tentacle-like appearance, lifelike skin, smooth motion and organic environmental setting, it was like no other robot ever constructed, yet it was expected to behave with convincing interactivity in brief engagements with people who had never met it before. More than any task-specialized or anthropomorphic robot, this robot triggered my particular interest in the design of bodily communication.

Leonardo, or Leo, (Figure 1-2) is the flagship robot of the Robotic Life group. Developed in conjunction with the animatronic experts at Stan Winston Studio, it was designed specifically

Figure 1-2: Leonardo, a robot designed for research into human-robot interaction.

for maximum physical expressiveness and communicative capability. Its intended use was as a testbed for multiple research threads — from character architecture development, to the testing of hardware initiatives such as motor controllers and sensate skins, to social AI and machine learning algorithm design, to fundamental HRI research involving user studies. It was therefore natural for me to become one of the students working on Leo as I transitioned from perceptual algorithms to overall communications architecture design.

Leo's computational architecture is derived from the ethological system developed by Bruce Blumberg's Synthetic Characters group. Having its origins in a system for controlling the behavior of graphical characters with simulated perception and the lack of real physical constraints, there were challenges involved in adapting this architecture to the uncertainties of the human-robot environment. However this simulation-based architecture was to prove invaluable when I chose to have the robot run real and virtual instantiations of its body simultaneously.

Several features of this body stand out. In terms of actuation, Leonardo is an upper torso humanoid robot, yet it is crafted to look like a freestanding full humanoid. Notwithstanding the lack of lower torso motion, with 63 degrees of freedom Leonardo is one of the most physically complex robots in existence. Most of these articulation points have been placed to facilitate expressiveness, and Leonardo is additionally extraordinary in being covered in fur as well as skin, to further influence this capability. The intentional design of an ambiguous combination of anthropomorphic and zoomorphic features serves to manage users' expectations and preconceptions, placing Leo's physical impression somewhere above the level of an animal but somewhere below the level of a full human, and offering no passive physical clues as to the character's approximate age.

Robonaut (Figure 1-3) is an upper torso humanoid robot developed by NASA in the Dextrous Robotics Laboratory at Johnson Space Center. It is being developed to be an astronaut assistant during manned space flight, and therefore has been designed to use human tools and equipment, and fit within the human space environment. It is approximately the same size and proportions as a human astronaut.

Our laboratory participated in work on Robonaut as part of a multi-institution collaboration during the DARPA MARS program. I was fortunate to be selected within our group to contribute

Figure 1-3: Robonaut, a robot designed for autonomous teamwork with human astronauts in space.

a component to the technology integration demonstration that would conclude the funding phase. The demonstration involved an interaction in which the human specified a task using deictic gesture. I had already developed a vision system for incorporating pointing gestures into interactions with Leonardo, but the way that the object referent was determined was quite simplistic, since the objects themselves were few and extremely distinct. The Robonaut collaboration gave me the opportunity to approach the issue of spatial reference communication more deeply.

Although Robonaut's physical characteristics are human-like in terms of body shape and manual dexterity, it has been designed with little concern for bodily expressiveness. Since astronauts wear bulky space suits and reflective visors, many aspects of bodily communication such as body language and facial expressions are simply not used. Instead, communication centers around spoken language and a tightly constrained gesture set, and thus this is the case with the robot also.

QRIO (Figure 1-4) is a full-body humanoid robot developed by the Sony corporation in Japan. It is designed primarily for entertainment and expressiveness through fluid bodily motion that resembles human movement. It has not been designed to replicate human morphology in other ways, however. QRIO's external shell has been designed to appear clearly robotic, and it does not simulate emotive facial expressions. QRIO is also much smaller than any human capable of commensurate coördinated motion performance.

My opportunity to work with QRIO came about through an internship with Sony Intelligence Dynamics Laboratory set up by Professor Ron Arkin of the Georgia Institute of Technology. This program comprised four graduate students from robotics groups at different institutions in the USA and Japan — a kind of robotic cultural exchange — and I immediately jumped at the chance to participate. One of the first questions Dr Arkin posed to me was, "What would you do if Leonardo had legs?"

Indeed, QRIO's most salient physical attribute is that it can locomote via the natural human means of bipedal walking. It is also able to vary its stride, take steps in any Cartesian direction, and turn in place. A further capability which was to prove particularly important is its ability to rotate its torso independent of its legs. This physical embodiment enabled me to explore the coördination of human-robot communication involving personal space, expressed using the same

Figure 1-4: QRIO, a humanoid entertainment robot capable of walking, dancing, and other expressive motion.

morphological modalities as humans do, which otherwise would not have been possible to include in this thesis.

1.4 Contributions

The contributions of this thesis are:

i. A computer vision based imitation game interaction for allowing a humanoid robot to rapidly generate a correspondence map between its body and that of a human teacher, without the need for specialized knowledge of motion capture algorithms or skeletal calibration mechanisms on the part of the teacher. The interactive nature of this mapping process allows the human teacher to exercise judgement on the correct appearance of the body correspondence, rather than relying on purely mathematical metrics that would be required in a fully automated scheme (Section 2.2).

ii. The design of a mental self-modeling method for the robot that uses additional instantiations of its cognitive structures to provide prediction and recognition capabilities based on time and arbitrary pre-chosen motion features. This self-modeling allows accurate predictive timing of action sequences, and further allows the robot to convert between motion representations of greater or lesser complexity, providing the flexibility of a task-independent, multiresolution approach to motion analysis (Section 3.3).

iii. The concept and implementation of inward attention, an analogue of perceptual attention but for introspective purposes. Practically, this model allows the robot to disambiguate communications containing context-dependent internal bodily references. More generally, it provides a framework for allowing the robot to reason about situations in which its own physical state is an important factor in the shared knowledge of an interaction (Section 3.4).

iv. A minimal communication framework, in terms of a suggested definition of minimal HRI awareness, for large-scale human interaction with a physically unusual, lifelike robot. A demonstration of this framework is provided in the form of an installation, based around real-time visual perception, that was exhibited for a week in a high-volume, semi-structured conference environment (Chapter 4).

v. The concept and implementation of a multimodal gestural grammar for deictic reference in which gestures as well as speech fragments are treated as grammatical elements. Benefits of this design are algorithmic component modularity, temporal gesture extraction, *pars-pro-toto* and *totum-pro-parte* deixis through the specification of hierarchical object constraints, and robust simultaneous referent determination in the presence of multiple identical objects within the margin of human pointing error (Chapter 5).

vi. The formulation of the "behavioral overlay" model for enabling a behavior-based robot to communicate information through encoding within autonomous motion modulation of its existing behavioral schemas. Schemas refer to an organized mechanism for perceiving and responding to a set of stimuli; they are the behavioral equivalent of an object in object-oriented programming. Behavioral schemas thus consist of at least a perceptual schema and a motor schema, and are typically organized hierarchically. The principal benefit of the behavioral overlay model is that it allows communication design to proceed largely independent of behavioral schema design, reducing resultant schema complexity and design overhead (Section 6.2).

vii. The design of a comprehensive proxemic communication suite for a humanoid robot, based on human proxemics literature, and its implementation on such a robot using behavioral overlays (Section 6.4).

viii. The design of a representation for precisely timed robot action sequences based on interval algebra networks, including extensions to the network representation to allow precise relative timing and context-dependent rearrangement of node clusters, and the implementation of this design within the `C5M` architecture. Contributions to the `C5M` architecture are the interval algebra network computation functionality, Self-Timed Actions, and the Scheduled Action Group encapsulation (Section 7.2).

ix. Development of the Style Axis Model for a robot, to enable a human director to request stylistic modifications to actions within timed sequences without necessarily knowing in advance the robot's internal motion representations nor the preëxisting style labels, and the implementation of this model within the `C5M` architecture. Contributions to the `C5M` architecture are the Action Repertoire prototype database, the Labeled Adverb Converter interface, Overlay Actions and the Speed Manager (Section 7.3).

x. The design and implementation in `C5M` of real-time, purely interactive multimodal communications sequence transfer from a human to a robot via a tutelage scenario. A human can instruct the robot in a multimodal communication sequence, visually verify its progress at each step, and then refine its timing and style through a synthesis of speech, gesture, and tactile stimulation. Design contributions include a style symbol grounding mechanism with theoretically optimal performance and the integration of

mental simulation and inward attention, and contributions to the C5M architecture include the Direct Joint Motor System, the Backup and Duplicator action interfaces, and action encapsulations for dynamically generated behaviors such as lip synchronization and eye contact (Section 7.4).

1.5 Thesis Structure

Since each of the robots involved in this thesis had unique physical characteristics that especially enabled the examination of particular kinds of communication, the thesis itself is to some extent naturally compartmentalized by both aspects. Rather than try to force generalizations, I have attempted to embrace this in structuring the thesis, so that each chapter represents a logical chunk that I hope can be digested both individually and in the context of the overall premise of human-robot communications design.

There is of course some overlap between the capabilities of the robots, so the chapters are not strictly separated according to the robot used, but since the architectural details are entirely robot-specific I have tried to keep these clearly delineated in the relevant chapters in order to prevent confusion. Similarly, rather than attempt to summarize all relevant literature in a single survey chapter, I have included within each chapter a focused literature review that places the chapter's work in the precise context of the related research efforts in robotics and other fields.

The thesis is also deliberately structured in the order that the chapters are presented. The first chapters represent infrastructural and design concerns that are intended as much as possible to be generally applicable, though of course they must make reference to specific implementations that I have designed on a particular robot. These are followed by chapters that focus more specifically on the design of certain types of communication, which are tied more inseparably to the robots whose special capabilities were used to realize these designs.

Chapter 2 contains a closer examination of the concept of common ground between a human and a robot. The term is defined and expanded into different logical types that may be useful in human-robot communications design. Since the primary salient feature of human-robot communication is the body, this chapter expands specifically upon bodily common ground, and presents a way in which it can be achieved through a brief, natural interaction with the robot.

Chapter 3 continues the general focus on the body, concentrating now on that of the robot and ways in which it can use self-modeling to improve its understanding of its own physical presence. This is placed in the context of the supportive groundwork such self-awareness can provide for various social learning methods. In particular, this chapter introduces the concept of an inward-directed self-attention system for disambiguating salient features of the robot's physical introspection.

Moving on from these more general chapters to specific types of communication begins with Chapter 4. In this chapter, the concept of minimal communication is introduced, referring to a baseline level of communicative performance in order to produce convincing interactivity. This is illustrated via a physically unusual robot performing in interactive robot theatre. This chapter includes a detailed description of the perceptual systems that supported the robot's minimal communications capabilities in a largely unconstrained setting.

In Chapter 5 the frame of communicative reference is expanded beyond the human and the robot themselves to include the spatial world and the objects within it. The focus is on the use of

multimodal reference to achieve spatial common ground between the human and the robot. This is accomplished with a combination of deictic gesture and speech, and the chapter describes a system for synthesizing the two into a multimodal grammar that allows the spatial arrangement of the environment to be simplified for the convenience of subsequent unimodal reference.

Chapter 6 expands upon gestural communication into a broad examination of non-verbal communication expression, including a comprehensive investigation of the use of proxemics. Since body language expression can be thought of as an adjunct communications channel to an individual's instrumental activities, a model is designed and presented for encoding this channel over a behavior-based robot's pre-existing behavioral schemas. This model is then demonstrated via application to a humanoid robot that can express proxemic communication partially through legged locomotion. The chapter also contains a description of the design of long term relationship and personality support to provide the robot with data for generating proxemic communication exhibiting rich variability.

Chapter 7 then closes the interactive communications feedback loop by describing how human tutelage can be used to shape a communications sequence through a process of reciprocal influence. A mechanism is described whereby a human teacher can direct a robot actor through a scene by assembling a sequence of component actions and then interactively refining the performance. The modification of the sequence concentrates on subtle expression, primarily through timing and motion style. A number of extensions to an ethological robot architecture are presented, including precisely timed sequences, models for style grounding, and management of the robot's known actions.

Finally, Chapter 8 presents the conclusions of the thesis and a summary of the evaluations of its designs, along with comments on the progress of human-robot communications and the future directions of the field.

Chapter 2

Common Ground

In a strong sense, communication is all about common ground. Consider an illustration, by way of contradiction, comprising a collection of agents that share no common ground. Each agent can perceive the others, but are completely unable to understand them or make themselves understood. Clearly, though perceptual information can pass from one agent to another via their observable responses, no actual communication can successfully take place — the scenario is akin to a vast, alienating corollary of Searle's Chinese Room thought experiment[266]. However, if the agents share some common ground, communication can take place; and we see from biology that if the agents are intelligent, this communication can then be used to share information that can lead to more accurate common ground, in a positive feedback loop. Common ground is therefore crucial to human-robot communication.

To read and understand the paragraph above, of course, one must already have a notion of what is meant by the phrase "common ground". This piece of understanding is itself a form of common ground, and this introductory text has been deliberately crafted in this circular fashion to illustrate that people, through their social development and biology, have a pre-existing folk psychological understanding of the general sense of the term. It is understood that common ground refers somehow to the grounding of the communication — ensuring that the communicating parties are "on the same page", that they are communicating about the same abstract things. The goal of common ground, it could be said, is to eliminate misunderstandings, and consequent frustrations, by providing an implicit set of disambiguous groundings.

But what does the term "common ground" really mean? According to the cognitive linguistic literature, common ground is the shared knowledge between individuals that allows them to coördinate their communication in order to reach mutual understanding[63]. This offers a sort of existence test, but of course it is possible to have more or less useful and appropriate common ground. A more analytical sense of the term is given by the postulate of least collaborative effort, which states that the optimal common ground for communication minimizes the collective effort required for a set of individuals to gain this mutual understanding[64].

In this chapter, I introduce common ground for human-robot communication, use psychological inspiration to cast it in the light of the mental duality between the self and the perceived other, and discuss general robotic implications. I then divide common ground for robots into three rough categories — that which most applies to the robot's body, to its shared environment, and to the

structures of its mental representations, algorithms and behaviors — and explain how these are addressed by the components of the remainder of this thesis.

2.1 Common Ground for Communication

During human-human communication, the participants automatically have a substantial amount of common ground as a result of their shared physiology and social development, even if this development involved significantly different enculturation. Perhaps most importantly, this common ground gives communicating humans an implicit idea of who knows what, and therefore the ability to estimate the minimum necessary amount of communication according to the postulate of least collaborative effort. Indeed, Lau et al. showed that the way people construct communications is strongly affected by their estimates of one another's relevant knowledge[167].

This is the case because humans construct a mental model of one another, incorporating their perceived abilities and intentions. Dennett theorizes that this is because humans (and other intelligent agents) are complex systems that cannot be explained and predicted merely in terms of their physical and design stances, although these are important; in addition, an intentional stance must be considered, reflecting the assumed presence of an internal mechanism that reasons about hidden state in the form of beliefs and desires[83]. As Braitenberg detailed in his thought experiment concerning synthetic vehicles[33], even relatively simple robots can exhibit behavior that can be interpreted in terms of an intentional stance. It therefore follows that robots must be designed in such a way that humans can construct an appropriately predictable mental model of them in terms of their abilities and intentions.

This can be illustrated with a familiar example from currently widespread technology, the voice telephone menu system. Encounters with such systems are renowned for being exquisitely frustrating, because they exhibit an almost complete lack of common ground. The user must communicate what he or she desires in absolute, sequential detail, and even then it can be difficult to be sure that the system has successfully achieved an accurate mutual understanding. It may therefore be valid to theorize that the minimal communicative effort translates quite well into the minimum resulting annoyance, at least for a given level of assured mutual understanding.

The converse applies to the robot as well. Common ground does not prevent miscommunication; according to the optimality definition above, it merely minimizes the amount of total communication required to resolve the situation. Therefore a robot that communicates effectively should also be able to assist in repairing misunderstandings. A strong way to support this is to model the human's perspective and context, in order to detect where the miscommunication might have occurred. In other words, achieving common ground means that the robot also needs to be able to develop compatible mental models of itself and its communicative partners. Development of a self-model is discussed in Chapter 3; here I discuss the background concerning the modeling of others.

In the psychological literature, such a model of the mental state of others is referred to as a theory of mind[241]. One of the dominant hypotheses concerning the nature of the cognitive processes that underlie theory of mind is Simulation Theory[75, 111, 126]. According to this theory, a person is able to make assumptions and predictions about the behaviors and mental states of other agents by simulating the observed stimuli and actions of those agents through his or her own behavioral and stimulus processing mechanisms. In other words, we make hypotheses about what

another person must be thinking, feeling or desiring by using our own cognitive systems to think "as if" we were that other person.

In addition to being well supported, Simulation Theory is attractive as a basis for the design of the mental architecture of a communicative robot because it does not require different models for the mental state of the human. Instead, the robot could represent other people from instantiations of its own cognitive system, assuming that the robot is indeed enough "like" the human through body and psychology.

In order to explain this capacity, a recent alternative to amodal representationalist theories of knowledge access has been suggested. This takes the form of treating cognitive experiential systems not just as recognition engines for sensory representations, but also as "simulators" that produce sensory reenactments or "simulations" which represent both recollected experiences and potential experiences that have not taken place, as well as implementing "situated conceptualizations" of abstract concepts that support pattern completion and inference[19].

This theoretical work provides an inspiration for designing robots to have internal structures that provide sufficient common ground to support communicative approaches based on self-simulation. Mechanisms for actually coördinating the re-use of the robot's cognitive modules in this way are introduced in Section 3.3. This chapter focuses on the underlying framework that allows such mechanisms to operate.

2.1.1 Related Robotics Work

The first to explicitly suggest that a humanoid robot required a theory of mind was Scassellati [260]. A comprehensive summary discussion of the need to examine the design and evaluation of human-robot interactions from the robot's perspective as well as the human's is presented by Breazeal in [41]. A complementary summary argument for increased attention to developing common ground in human-robot interaction, including more detailed descriptions of some of the investigations mentioned below, is given by Kiesler in[150].

A number of experiments have demonstrated that the insights from human-human communication related to common ground are also valid for human-robot communication. Powers et al. investigated aspects related to least communicative effort, demonstrating that people querying a robot engage in less total communication under conditions in which they think the robot is experienced in the subject matter[240]. The precise nature of the robot's experience is also subject to human extrapolation based on externally perceivable common ground. For example, Lee et al. showed that the national origin of a robot affected people's assumptions about what landmarks a guide robot was familiar with[170].

These results suggest that the converse situation should also be considered, concerning the nature of the robot's deliberate communications. If minimal effort communication principles apply to common ground between humans and robots, it supports the contention that in order to forestall annoyance, the robot should use mental modeling techniques based on external perceptions to try and infer as much as possible about the human rather than pestering them for the information via explicit communication (assuming that this information can be obtained accurately).

It seems also to be the case that the nature of the mental models humans construct is situation-dependent. Goetz et al. showed that the correspondence of robots' appearance and behavior to

the task it is expected to perform affects the degree of human-robot coöperation that results[110]. Similarly, researchers at the Institute for Applied Autonomy showed that performance could be improved by using an appropriately-designed robot rather than an actual human in a communicative task[133]. These results suggest that the robot should therefore also be able to take into account the environmental context of the interaction as part of its common ground representation.

Common ground pervades ongoing work in our laboratory. Examples of research involving these principles includes the management of expectations in human-robot collaboration[128], using a simulation theoretic approach for achieving cognitive common ground in terms of inferred goals[113], and of course this thesis itself. In the remainder of this chapter I categorize the types of common ground that have been considered, and explain how they relate to the work that has been performed.

2.2 Bodily Common Ground

Before any other considerations are raised, a human and a robot already have one thing in common: they both have a body. Thus it is possible to roughly isolate a class of common ground for communication in terms of shared aspects that the robot can ground to its own body. In other words, for the robot to theorize about the physical context of the human's communication, it needs to be able to understand the communicative aspects of the human's body. If the human and the robot share no physical common ground other than their embodiment itself, the situation is one of minimal communication, approached in Chapter 4. If the human and the robot do share common aspects of external physiology, the most simulation theoretic way to represent this is to attempt to relate the human's body to its own.

As with other elements of this chapter, parallels can be drawn from this to physiological theories. Recent neuroscience research indicates that primate brains may indeed contain a dedicated neurological apparatus for performing this body "mapping". It has been found that the same collection of neurons in the primate brain, known as mirror neurons, become active when an individual observes a demonstrator performing a certain task as when the individual performs the task itself[248]. Results have similarly been reported from human studies[132].

In order to achieve bodily common ground with our humanoid robot Leonardo, we[1] have implemented a body mapping mechanism for inspired by the function of mirror neurons. Because we favor a developmental approach over manual modeling, the robot learns the body mapping from interaction with a human teacher. The vision-based human motion data is provided in the form of real-time 3-D position and rotation information of virtual bones in an articulated, simplified human skeleton model driven by a Vicon motion tracking system. This must be mapped into data that can drive the robot's own motion, in the form of the rotation of the various actuators that move its skeleton.

Traditional machine learning solutions to this problem exist, but are time consuming and tedious to apply. Reinforcement learning is one option; the robot could attempt to imitate the human, and the human could reward correct behavior. A domain like this, however, has an extensive search space, and constant human supervision would be required to detect correct results. Also,

[1]The body mapping procedure was first implemented by Jesse Gray, using an electromechanical "Gypsy suit" to record the motion of the human. Matt Berlin assisted with linear interpolation functions on one of the later vision-based implementations.

there is the potential to damage the robot in the early stages of the process, as it tries to imitate the human with a flawed body mapping. Many supervised learning algorithms exist which use labeled training data to generate a mapping from one space to another (e.g. Neural Networks, Radial Basis Functions). These algorithms tend to require a large number of examples to train an accurate mapping, and the examples must be manually labeled. To apply such algorithms to this problem would require many examples of robot and human poses which have been matched together such that each pair contains robot pose data that matches the associated human pose data. However, acquiring and labeling that training data through typical means would be quite tedious and time-consuming. Instead, this tedious interaction has been transformed into an intuitive game played with the robot.

In order to acquire the necessary data to teach the robot its body mapping, a simple but engaging "do as I do" game was developed, similar to that played between infants and their caregivers to refine their body mappings [196]. Acquiring bodily data in this way may then be taking advantage of an innate desire humans have to participate in imitative games. The imitation interaction is simple: when the robot acts as leader, the human attempts to imitate it as it adopts a sequence of poses (Figure 2-1). Because the human is imitating the robot, there is no need to manually label the tracking data — the robot is able to self-label them according to its own pose.

Figure 2-1: Robot and human playing the body mapping imitation game.

As the structure of the interaction is punctuated rather than continuous, the robot is also able to eliminate noisy examples as it goes along, requiring fewer total examples to learn the mapping. When the robot relinquishes the lead to the human, it begins to attempt to imitate the human's pose. The human can then take a reinforcement learning approach, verifying whether the robot has learned a good mapping or whether a further iteration of the game is required. The entire process takes place within the context of a social interaction between the human and robot, instead of forcing the robot offline for a manual calibration procedure which breaks the illusion of life, and requiring no special training or background knowledge on the part of the human teacher.

It is diffult for the teacher to imitate the entire robot at once; when imitating, humans tend to focus on the aspect they consider most relevant to the pose, such as a single limb. Furthermore,

humans and humanoid robots both exhibit a large degree of left-right symmetry. It does not make sense from either an entertainment or a practical standpoint to force the human to perform identical game actions for each side of the robot. The robot's learning module is therefore divided into a collection of multiple learning "zones". Zones which are known to exhibit symmetry can share appropriately transformed base mapping data in order to accelerate the learning process.

The Vicon tracking system models the human skeleton as an articulated arrangement of "segments" (roughly equivalent to bones) connected by kinematically restricted joints. However, although the kinematic restrictions are used to constrain the tracking, the real-time output data stream does not contain the resulting joint angles inferred by the tracker. Instead, each segment was described with an independent 3-DOF rotation. It was therefore necessary to perform a kinematic decomposition of the tracking data prior to using it as input to the body map training algorithm.

In order to provide the training algorithm with data whose dimensionality closely matched that of the robot's joint space, this "kinematicator" modeled the human skeleton in terms of standard orthogonal-axis rotational joints. For example, the wrist is modeled as a 2-DOF Hardy-Spicer joint, and the elbow as a 1-DOF hinge joint. Similarly, to model segment rotations such as the human radius-ulna coupling, certain segments such as the forearm were allowed their own axis-centered rotations.

When the kinematicator is run, it decomposes the tracking data into dependent joint angles, and sets warning flags in the event that joints exceed certain generous limits (such as reverse bending of the elbow hinge). These joint angles are then fed into the training algorithm, which uses a radial basis function (RBF) approach that effectively captures interdependencies between the joints and provides faithful mapping in the vicinity of the example poses. Once this non-linear mapping is trained, it can then be used in real time to map tracked human joint values onto joint angles on the robot. This process underpins various communication-related capabilities used in this thesis, such as self-modeling of imitation (Section 3.3.2), inward self-attention (Section 3.4), and action recognition during interactive communication shaping (Section 7.4.1).

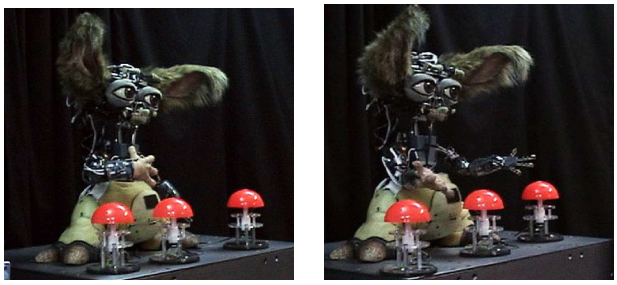
2.3 Spatial Common Ground

During a copresent interaction, another thing that the robot and the human automatically have in common is the fact that they are both spatially situated in the environment. Furthermore, there are likely to be other things that are also spatially situated along with them, such as objects. It is therefore possible to isolate another rough area of communicative common ground from which the robot can seek to draw knowledge to support mutual understanding: things that it can ground to the physical world. The robot should endeavor to ensure that it shares a common spatial frame of reference with the human, and that it physically utilizes space in a fashion that does not cause communicative misinterpretation on the part of the human.

These aspects can be considered separately. A common spatial frame of reference concerns a shared understanding of which objects are where, including the human and the robot, which of them are spatially important, and what their spatial relations mean to the interaction. Joint visual attention is one of the mechanisms underlying the creation and maintenance of such a common frame of reference[259, 202]. Our robot Leonardo uses and interprets spatial behavior

Figure 2-2: The robot acknowledges the human's spatial reference to an object by confirming joint visual attention through the direction of head and eye pose towards that object.

to influence joint visual attention, such as confirming object attention with head pose (Figure 2-2), and distinguishing between itself and the human with deictic gesture (Figure 2-3). Part of this thesis specifically addresses more comprehensive communication about and within a common spatial frame of reference, which can be found in Chapter 5.



(a) Leo indicates himself. (b) Leo indicates the human.

Figure 2-3: Communicative spatial gestures from the robot.

The communicative content of personal space utilization is known as proxemics[120]. Although this has largely been neglected in the HRI field to date, it is a valid component of spatial common ground because humans ascribe emotional and intentional causality to observed features of interpersonal distance and posture. Robots with spatial mobility therefore can interfere with communication by being unaware of proxemic considerations, even if they do not explicitly attempt to use proxemics as a communication mechanism. Furthermore, personal space considerations vary widely across human cultures, meaning that interpersonal common ground is likely to vary from human to human and thus should be modeled by the robot on this basis. This topic is considered in detail in this thesis, as part of the component on non-verbal communication in Chapter 6.

2.4 Cognitive Common Ground

The last category of common ground is the most broad, because there are no further physical attributes that can be used for direct reference. It is cognitive common ground, referring to shared information about the mental representations that the human and the robot have within them. This is also the most difficult area, because the degree of commonality between those representations is an open question. Certainly the precise representations will not be the same, given that the way human mental states are represented in their connectionist biology is far from known, and robots currently possess impoverished mental representations implemented on digital computers. Nevertheless, since much communication concerns these cognitive aspects — goals, motivations, attitudes, relationships, emotions, language symbols, and so on — the robot must endeavor, through design and learning, to ground its understanding of the human to appropriate cognitive representations.

Some common ground in this area can often be assumed by considering zoomorphic cognitive aspects that are shared by almost all intelligent autonomous agents that humans generally encounter. For example, behaviors such as aggression and fear in living systems are generally accompanied by relatively unambiguous communicative mechanisms. These ubiquitous mental representations support baseline communication capabilities that are examined in Chapter 4 on minimal communication.

In the case of most robots designed for human interaction, a straightforward design approach has been taken to facilitating analysis and attainment of cognitive common ground. The work in this thesis on non-verbal communication, in Chapter 6, is a good example of this. The focus of attention has been on the algorithms for accurate expression of mental representations in order to maintain common ground through external observability; however the mental representations themselves have been carefully designed and tuned to make cognitive common ground innately feasible in the first place. The internal state descriptors of the robot are deliberately human-like, and the output expressions themselves are based on published results from exhaustive scientific observation of human communications.

Similarly, the use of symbolic linguistic elements to achieve common ground across abstract concepts, as the system for interactive manipulation of physical communication style does in Chapter 7, tends to be explicitly designed with human-like representations because the Symbol Grounding Problem is an unsolved problem[124]. However, this is not a criticism of the practice of designing robotic systems to have mental representations that reflect those of humans (or at least the way humans introspectively consider things to be represented). On the contrary, I argue that this is essential to effectively coördinating communication between humans and robots.

At this point I will briefly describe two related pieces of this work that are not detailed elsewhere in the thesis, but that are relevant to the area of cognitive common ground. The design of the robot's processing systems to have commonality with human mental representations is necessary, but there are clearly limits to how far this can be successfully achieved. The robot must also have mechanisms that allow it to adapt to aspects of human cognition that it learns from its interactions with humans. In particular, if it already has sufficient cognitive ground to effectively receive specific guidance from humans concerning rules for interpreting human goals, motivations etc., it should ensure that such instruction is facilitated. I have therefore made a couple of additional forays into specifically enabling and motivating the human teacher to transfer these constructs.

An early interaction with Leonardo involved the use of a set of buttons that both human and

robot could manipulate between states of ON and OFF (Figure 2-4). Interactions could therefore be described in terms of goals such as changing the state of the set of buttons, or simply toggling a button. Certain aspects of cognitive common ground in this interaction were innate, such as the states of the buttons themselves, and the basic goal types. However, compound goal-directed tasks involving these goals types could be taught to the robot by the human.[2] Once a task had been defined and requested, the robot would display a commitment to executing it until it was observed to be complete.

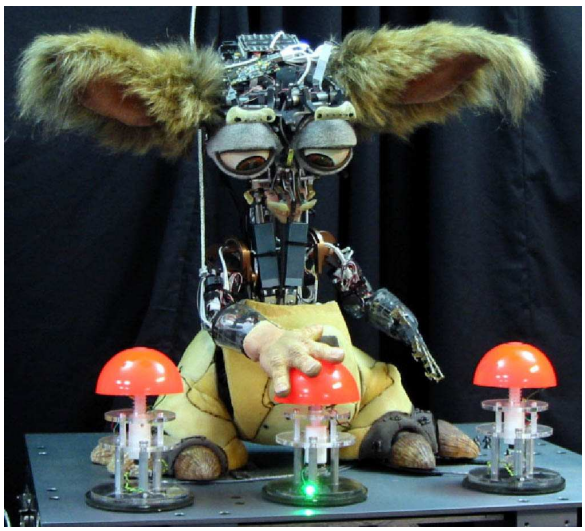


Figure 2-4: Leonardo executing a human-communicated buttons task.

I extended this scenario to incorporate the concept of disjoint task goals, such as a competitive game. The robot used its existing task representation to achieve common ground with the human's cognitive activity by allowing the human to inform the robot that he or she was committed to a separate task. The robot would then display commitment to its own task only as long as neither of the tasks had been completed. Details of this interaction were presented in [52].

I also experimented with a physical therapy training interaction that made use of cognitive common ground in the form of emotional feedback to motivate the human trainee into performing an exercise that the robot could then use as a data source for analysis of improvement or degradation in the human's condition. Physical common ground was first used to allow a different human caregiver to imitatively communicate to the robot a physical exercise to be performed with the patient. Thus cognitive grounding in the form of the goal itself (a specific physical movement) was transferred from the human to the robot, but the general goal structure (some physical movement) was innate.

The robot could then demonstrate the exercise task to the human patient, and proceed to perform the exercise in concert with him or her, while giving straightforward emotional feedback in order to motivate the patient and communicate the state of the exercise interaction in addition to various modes of physical feedback concerning the exercise itself (Figure 2-5). Thus once again the cognitive representations for supporting common ground with respect to the goal and the

[2]The goal-directed buttons interaction was primarily developed by Andrea Lockerd.

Figure 2-5: A subject performing a physical therapy exercise in concert with a graphical representation of Leonardo (inset).

motivations are innate (e.g. human-like emotions), but the specific details of the goal and the human's performance feedback were able to be communicated. This system was originally described in [50].

Chapter 3

Bodily Self-Awareness

The establishment of common ground sets the stage for effective communication, as discussed in the previous chapter. Nevertheless, human communications are complex, and to design robots for human interaction requires careful attention not only to the modes of communication but their purpose. Among the things that can be communicated are the states of the participants, details about the world and the objects in it, and aspects concerning the interaction itself. All of these are covered in later chapters. However there is one important area of communication which requires additional support. This area is the communication which underpins social learning, and the support to be discussed here is special information that the robot can benefit from understanding about its own self.

The robot, of course, has access to plenty of data about itself. From low to high abstraction and back, it has sensor values, perceptions based on these values, beliefs about the world based on these perceptions, actions currently being triggered based on these beliefs, and ultimately motor positions and velocities generated by these actions. Being a real robot, all of these are situated in and grounded on the real world. From Chapter 2, insofar as the real world contains humans, the robot is able to relate certain aspects of the behavior of those humans to itself. However, there is another side to this coin. Humans have a highly developed theory of mind. They have an introspective understanding of their own minds and bodies, and their causal relations with the world, and they attribute similar understandings to other intelligent agents within the world. If the robot presents itself as an intelligent agent, the human is likely to expect it to have some understanding of itself and the internal context of its own behavior beyond its mere responses to stimuli and internal drives.

For an example of what is meant by this, consider a typical algorithm for motion planning and obstacle avoidance that uses knowledge of the robot's physical extent in order to ensure a valid solution. The robot's data concerning its own shape and size does not constitute self-awareness in the sense used here unless it can relate this information to a non-static context in which its actions occur. Consider, for example, a robot that can reconfigure its body in order to navigate differently shaped passageways. A human instructs it to squeeze itself down and go to the next room, which is accessible by both a large and a small passage. A robot without self-awareness might treat the commands separately, making itself more compact and then planning a path independent of its new configuration. In contrast, a robot with bodily self-awareness could be expected to inform its path

planning algorithm based on the context of its change in configuration, and thus be more disposed to select the passage implied by that action, namely the smaller one.

In this discussion of self-awareness, it is important to contrast what is meant by this term as opposed to the phenomenon of consciousness. The nature and explanation of consciousness is an unsolved problem, and for a sampling of divergent views I direct the reader to the Sensorimotor Theory of Consciousness[221] and the Unconscious Homunculus[72]. Rather than get bogged down in this debate, I am concerned simply with the practical management of expectations in the interaction. When a human is attempting to instruct the robot through socially guided learning, it is likely to be frustrating and unmotivating if the robot does not exhibit a basic sense of knowing what its body is doing. In other words, it needs to be able to represent physical aspects of itself in such a way to facilitate introspective analysis, regardless of the precise form in which such a representation is manifested. In order to facilitate high-level social learning, we must say: "Robot, know thyself".

In this chapter, I start with an overview of some socially interactive learning techniques that can benefit from addressing the issue of bodily self-awareness, and some related work that has made use of these techniques in robotics. Then I present three aspects of bodily self-awareness that I have examined and implemented in order to support human-robot communication. First, I look at scaffolding imitation through physical contact and reflex actions. Second, I present a mechanism for using the robot's own mental structures as a simulator for visualizing, reasoning about, and making predictions concerning its own body. Third, I tie this all together by augmenting the robot's attentional capabilities with a system for facilitating the direction of attention inwards to itself rather than just outwards to the environment.

3.1 Social Learning and the Self

Socially situated learning has in recent years become an attractive topic in robotics research, in terms of its potential for leading to new mechanisms for endowing robots with additional competencies. This is primarily due to the "existence proof" offered by human and other primate systems. In his taxonomy of socially mediated learning, Whiten identifies a number of mechanisms of social transmission of behavior that occur in these groups, such as imitation, emulation, and significance learning (including stimulus enhancement)[302]. To this, since we can incorporate high-level communicative interfaces into the robot, we must also add direct tutelage. All of these examples can benefit from bodily self-awareness.

Whiten reserves the bulk of his taxonomic discussion for imitation and emulation, and this focus is also reflected in the work performed in the robotics community at large (see below). Imitation is the process by which the learner observes repeated correct demonstrations of a task, and modifies its own behavior until it achieves comparable results[103, 314]. In many cases, these results refer to the accomplishment of an observable effect on the world[57]. The observability of the result is key, because that is the mechanism which the learner uses to determine when the task has been adequately learned. It therefore follows that if the desired results are specific to the embodiment of the learner — for example, a dance move, or a yoga position — in order to learn them imitatively, the learner must have a means of observing its own physical behavior and comparing it to the demonstration in order to generate a qualitative or quantitative error signal.

A perhaps more powerful form of imitative learning is what Tomasello originally names "emulation": "a focus on the demonstrator's goal may lead the observer to be attracted to and seek to obtain the goal"[285, 304]. Indeed, emulation — also known as "goal emulation" in an attempt to further disambiguate the concept[303] — has been the focus of much recent research, and comprises the bulk of the update to Whiten's ten-year-old taxonomy[302]. Despite its broader scope, emulation benefits from bodily self-knowledge in much the same manner as regular imitation. Once again, this concerns situations in which the involvement of the learner's body is instrumental in the goal state. For example, if the goal state is to possess a particular object, the learner must know what it means to physically possess the object (e.g. by grasping it) and thus whether or not it is doing so.

Breazeal and Scassellati identify a number of open questions with regard to imitation learning for robots, beyond those involved in providing the robot with the actual imitation skills themselves [47]. Perhaps the two most key such problems are *when* and *what*. The former asks how the robot knows when to imitate — both in terms of when an appropriate demonstration is occurring, and when the imitative behavior should commence and finish. The latter asks how the robot knows what to imitate — being able to determine what aspect of the demonstration is important, and what kinds of demonstrations are appropriately responded to with imitation.

Clearly both of these questions can not be answered without bodily self-awareness when the demonstrations in question concern bodily configurations. In particular, the robot needs to know whether its bodily state is compatible with the imitative behavior in question. In terms of the question of when to imitate, this might simply concern whether the robot is physically able to actually start imitating when its imitative system declares it desirable. The question of what to imitate is more fundamentally crucial, however — bodily self-awareness allows the robot to actually determine whether or not it can imitate the demonstration under any circumstances, and if so, to rule out aspects of the demonstration that are irrelevant by virtue of being physically incompatible with its body.

In contrast to mechanisms involving observation of deliberate or incidental demonstrations, other social learning methods can involve intentional manipulation of the learner's perceptions for pedagogic effect. Stimulus enhancement, or local enhancement, involves the direction of the learner's attention to particularly relevant aspects of the environment[275, 283]. Once again, it is easy to see that if an instrumental component of a teaching stimulus involves the learner's own body, then the learner must be sufficiently aware of this aspect of its own body to enable its attention to be drawn to it.

A particularly pertinent aspect of this technique is widespread among certain primates in teacher-learner interactions involving physical tasks. In these demonstrations, teachers will physically manipulate learners' bodies into the desired configuration, rather than performing the task themselves and expecting the learners to mentally map the relevant configuration. For example, in teaching the correct way to grasp a tool, the teacher forces the learner's hand around it in the correct position, so that the learner can learn from observation of its own sensorimotor experience. This could be described as a form of self-imitation, and clearly it requires a corresponding self-awareness of the body in order to make this observation possible.

Finally, the otherwise uniquely human capability for social learning via direct tutelage involving high-level dialogue can also depend on bodily self-awareness. High-level instruction frequently involves contextual ambiguity, and when this relates to the learner's physical embodiment that

physical context can be used to disambiguate the instruction. This ambiguity is not necessarily a direct result of task complexity, but rather is largely due to deliberate underspecification of the tutelage dialogue for efficiency and reduced tedium on the part of the instructor. For example, when teaching a manipulation task and referring to a specific placement of a hand, the grounding of the hand concept to an actual one of the learner's hands is dependent on a product of current resource usage and the symmetry of the task. Sometimes it does not matter which hand is used, and other times it is critical that it be a particular one. Bodily self-awareness on the part of the learner is necessary in order to resolve the bodily context of the instruction.

3.1.1 Related Robotics Research

As mentioned above, imitation learning is a popular research area in robotics, and good overviews of the field have been written by Breazeal and Scassellati in [47] and by Schaal in [261]. Many efforts have concentrated on the learning of physical motor skills and physics-based models to drive them. For example, robots have been taught via observational imitation to stack blocks[161], balance a pendulum[14], perform a tennis return swing[263], and to learn aerobic-style sequences[216].

Rather than the learning of specific physical tasks, this thesis is more concerned with supporting the use of imitation as one of the learning techniques underpinning a generalized natural socially guided teaching approach. Indeed, some of the prime researchers in this area have suggested that on the one hand imitation is part of a route to more natural human-robot interaction[214], and that on the other, interdisciplinary studies of social learning in robots versus natural systems can be a path to better understanding of both[213]. The more relevant related work is thus imitation and social learning material in which a more central aspect is the robot's awareness of its own behavior in the context of the environment.

At the lowest level, these involve giving the robot an internal model of its physical motor system that it can refer to in some way, for example to use in computing a metric of imitation performance. Some approaches to internal motor system representation are reviewed by Schaal et al. in [262]. There is some valid dispute about whether internal physical models should be closer to the real body of the robot, or to the real morphology of the teacher from which the robot must learn, or something in between. In the case of humanoid robots learning from humans, this may be less important, but convincing arguments have been raised in favor of intermediate physical representations in cases when demonstrations might be used to train robots having similar morphologies that nevertheless differ in important ways (e.g. being able to train a generic quadruped robot such as the Sony AIBO)[244].

Higher level internal models have similarly been proposed to facilitate higher level reasoning, such as action classification and prediction. Demiris and Hayes have championed the use of multiple inverse and forward models for this purpose[82]. More recently, hierarchical arrangements of these have been used for top-down control of attention during action perception, to more rigidly constrain the learning problem [148]. The system works by recognizing the demonstrated action with the inverse models, then using the forward models to direct the robot's attention towards the environmental variables that it would control if it was executing the observed action. This could thus allow the robot to concentrate on the important aspects of a demonstration intended to refine its ability to do something it already knows to some extent how to do. However the attention being referred to concerns external rather than internal features.

Some research has also been directed towards the use of physical scaffolding and the potential learning benefits of physical engagement in the demonstration. Some early work in training a robot arm to complete a physical goal made use of physical contact from the teacher in the form of "nudges" that helped it ultimately infer the intention of the movement[294]. More recently, the difference between active and passive observation of a demonstration for driving imitation learning has been investigated. During "dynamic observation", in which the robot simultaneously physically imitates some aspects of the demonstrated behavior while continuing to observe the demonstrator, the learner has a potential advantage from the increased data flow from its own sensorimotor system as well as from the sensory representations of the teacher's actions. This work has been performed on Khepera robots engaged in simple following behaviors[257].

Similarly, in our laboratory we have placed a great deal of emphasis on the paradigm of "learning by doing", in which the robot physically performs actions learned through social mechanisms such as tutelage and imitation[43]. This has primarily been motivated as a means of maintaining task common ground with the human teacher (Section 2.4), but can also be used to allow the robot's self-awareness as an information source in the process. For example, rudimentary emulation behavior in the presence of failed demonstrations has been demonstrated. The robot compares the human's action with known goal-directed actions in its own repertoire, determines whether or not the goal of the selected matching action was achieved, and then offers to perform its own action to achieve the goal if it deems that the human teacher "failed"[112]. The work in this chapter extends the use of the robot's bodily self-knowledge as such an information source.

3.2 Physical Reflexes for Self-Directed Stimulus Enhancement

Given the physicality of both the robot and the human with whom it is interacting, it is desirable to support the stimulus enhancement mode of physical communication that is observed among primates and was described earlier. That is, to allow the human to manipulate the robot's body into a desired position, and then to use a "lower-bandwidth" communication to trigger a learning response that takes into account that bodily configuration. It is likely to be more efficient and natural to teach a posture this way than to exhaustively describe it with other modalities such as speech.

Unfortunately, there are often complications preventing this procedure from occurring in the same way with robots as it can with animals. In the case of our humanoid robot, Leonardo, these difficulties concern the nature of its physical embodiment: the fact that its skeletal joints and actuators are made up of inelastic inorganic gears, motors and cables, rather than compliant organic bones, sinews and muscles (Figure 3-1). Put simply, the robot is not designed for external physical manipulation. Its geared servo-motors are not designed for backdriving, so its arms, for example, are likely to resist and ultimately break when pushed upon; its cable-driven fingers are compliant in one direction (towards the return spring), but not the other. It is possible to design robots with "soft" actuators that are intended to be pushed around by other agents in the environment, but for Leonardo another mechanism is needed.

Although Leonardo is not designed for physical joint manipulation, it is designed for touch interactions with humans, and we[1] have therefore equipped it with tactile sensors in places under

[1]Leonardo's tactile sensing apparatus was designed and built by Dan Stiehl. I incorporated the tactile data into the behavior system and designed these touch interactions.

Figure 3-1: The Leonardo robot without its skin installed, showing some of its skeletal and outer shell structure.

the skin. The mechanism that has been selected to allow humans to physically shape Leo's bodily configuration is therefore a set of tactile reflexes. Different kinds of touching trigger changes between various bodily states. The motor representation for these state changes is implemented with direct joint motor systems (Section 7.3.1). Ultimately the tactile sense capability is intended to be included under all of Leo's skin, but at present it is just included in the hands.

The actual tactile sensing capability is essentially a pressure sensing capability, accomplished with a number of force sensitive resistors (FSRs) located on circuit boards underneath Leonardo's silicone skin. Each hand has an asymmetric array of 9 FSRs on both the palm and the back of the hand, and 3 FSRs on the edge of the hand below the pinky finger. In addition, each finger has 5 FSRs on the distal segment: one on the bottom (corresponding to the finger pad), one on the top (where the nail would be on a human), one on each side, and one on the very tip. The layout of the hand sensors is shown in Figure 3-2. A dedicated microcontroller in each hand regularly samples the FSR values and sends them via a serial link to a monitoring computer. For more details on the design and construction of Leonardo's tactile apparatus, see [276].

The FSR values are normalized according to a precalibrated range, and are sent to Leo's behavior engine over IRCP (our proprietary Intra-Robot Communication Protocol). They are then decomposed into a set of hierarchical percepts. At the root of the hierarchy is whether either of the hands were touched, proceeding down to the specific areas of the hand and fingers, and then to the leaf nodes which determine whether or not a region was "pinched" (Figure 3-3). Touches are defined as instances when any normalized FSR value in a hand subregion exceeds a threshold, and pinches occur when two FSRs that are physically opposite each other (for example, the left and right sides of a finger) exceed the touch threshold. In addition to these derived percepts, a general tactile percept makes the raw normalized FSR values available to any perceptual subsystem that wishes to make use of them.

In Leonardo's regular action system (see Section 7.2.1), a "Reflex Action Group" is set up with

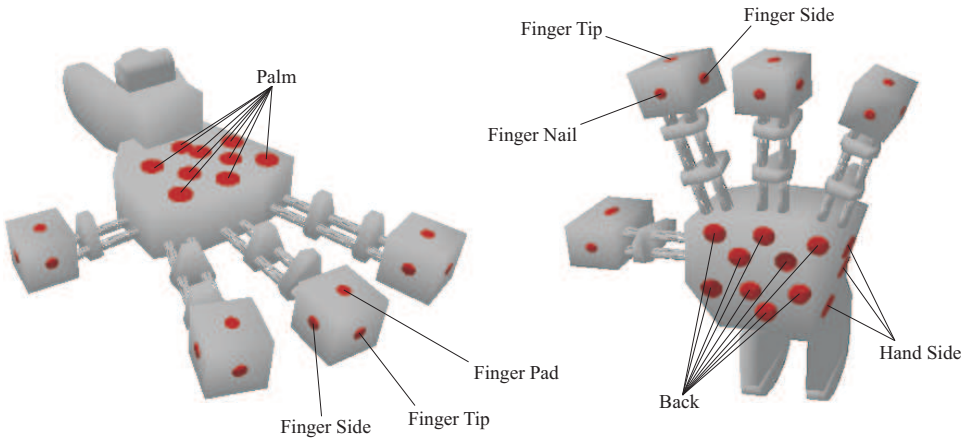


Figure 3-2: Layout of Leonardo's tactile sensors.

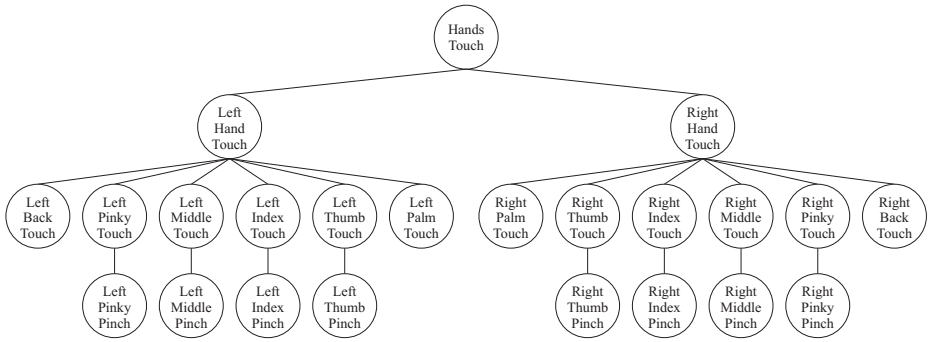


Figure 3-3: Leonardo's tactile percept hierarchy.

a collection of innate reflexes relating to the robot's tactile perception. The purpose of these is for the robot to react to external influence on its hands by predictably altering their configuration. For example, when the robot's palm is stroked, it reflexively curls its fingers in to grasp the object that is touching it. This action was inspired by automatic grasp behavior in infants.

Although the curling inward of Leo's fingers is in the direction of the finger return springs, and thus this configuration could be accomplished by the human forcibly compressing and molding the robot's hand into a fist, the force needed to overcome the return springs is so small that the sensitivity of the FSRs is insufficient to reliably determine when this is occurring. Reflexive opening and closing of single fingers is accomplished via the perception of pinching. When a finger is squeezed, the robot instinctively toggles its state. This is not a completely natural reaction, but since the force needed for reliable FSR discrimination is non-trivial, it is safest to do things this way rather than risk breakage of the synthetic cable that drives Leo's finger curl mechanism, which is not extremely strong. A graphical view of Leo's tactile reflexes is shown in Figure 3-4.

In order to make use of these reflexes, two things are needed. First is a structure to monitor and encapsulate bodily configuration, and second is a mechanism of deciding precisely which aspects of bodily configuration should be encapsulated. The latter is part of the inward-looking attention system that is described in Section 3.4. The former is essentially an abstraction over Leo's most basic internal representation of body position in terms of joint motor positions.

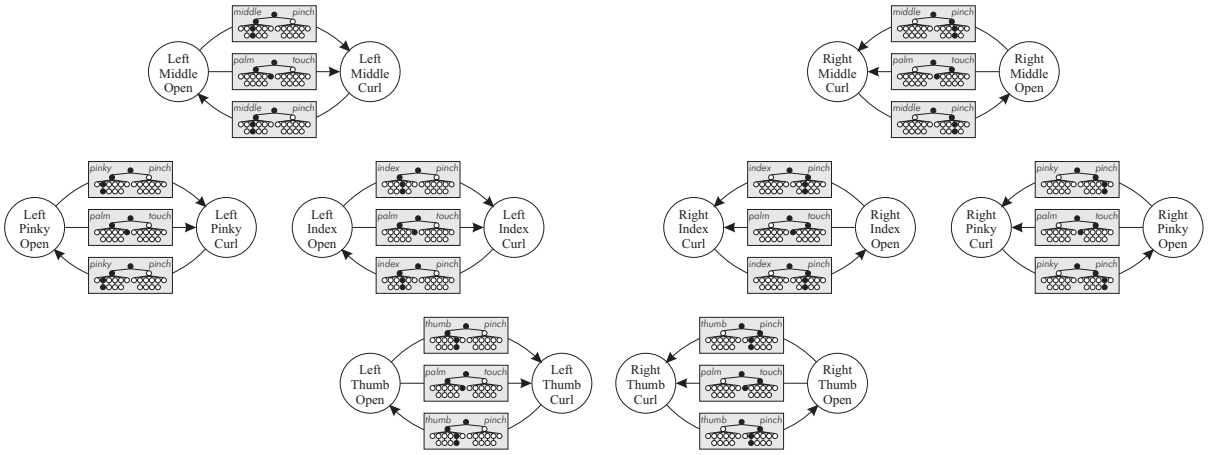


Figure 3-4: Leonardo's tactile reflexes.

A body state monitor was therefore developed that takes as input a bodily region and produces a "snapshot" of a subset of the region's joint motor values that can subsequently be used as a destination for a direct joint motor system. The selection of which joints go into the subset is made by the inward attention system, for example to differentiate between the right and the left hand. A sample interaction therefore goes as follows. Using interactive touch, the human manipulates Leonardo's hand into a desired pedagogic position, for example a pointing gesture, by physically triggering the robot's tactile reflexes. Once the robot's hand is in the correct position, the human issues a learning trigger, such as a spoken command, to cause an association in the robot's memory. The robot then treats its own bodily configuration as part of its "conscious" perceptual system, examining the state of its appropriate hand and adding it to its repertoire of known poses — in essence, "self-imitating" for future reference (Figure 3-5).

3.3 Mental Self-Modeling

As was discussed in this chapter's related work, having a mental model of one's own body can confer certain advantages. From an appropriately representative mental model a robot can reason about what its body *can* do, and make predictions about what it *will* do during physical behaviors. In this sense a mental model is like a self-simulator, allowing the robot to "imagine" itself doing the various things it can or might do. Leonardo has therefore been equipped with an "imaginary Leo" mental model that supports this kind of self-simulation.

The behavior engine that is used to control our robots is a modular architecture that is already ultimately based on an exact kinematic model of the robot. The lowest-level motor controllers in fact monitor the joints of this kinematic model and use that data to drive the actual robot. In this sense, the robot could already be said to have a mental self-model. However, it is necessary to distinguish between an internal self-representation and a true mental model, because under ordinary operating conditions this representation cannot be used for simulation, because any activity occurring to it is automatically reflected into physical output on the real robot. Nevertheless, the existence of this kinematic model simplifies the process of creating a true mental model.

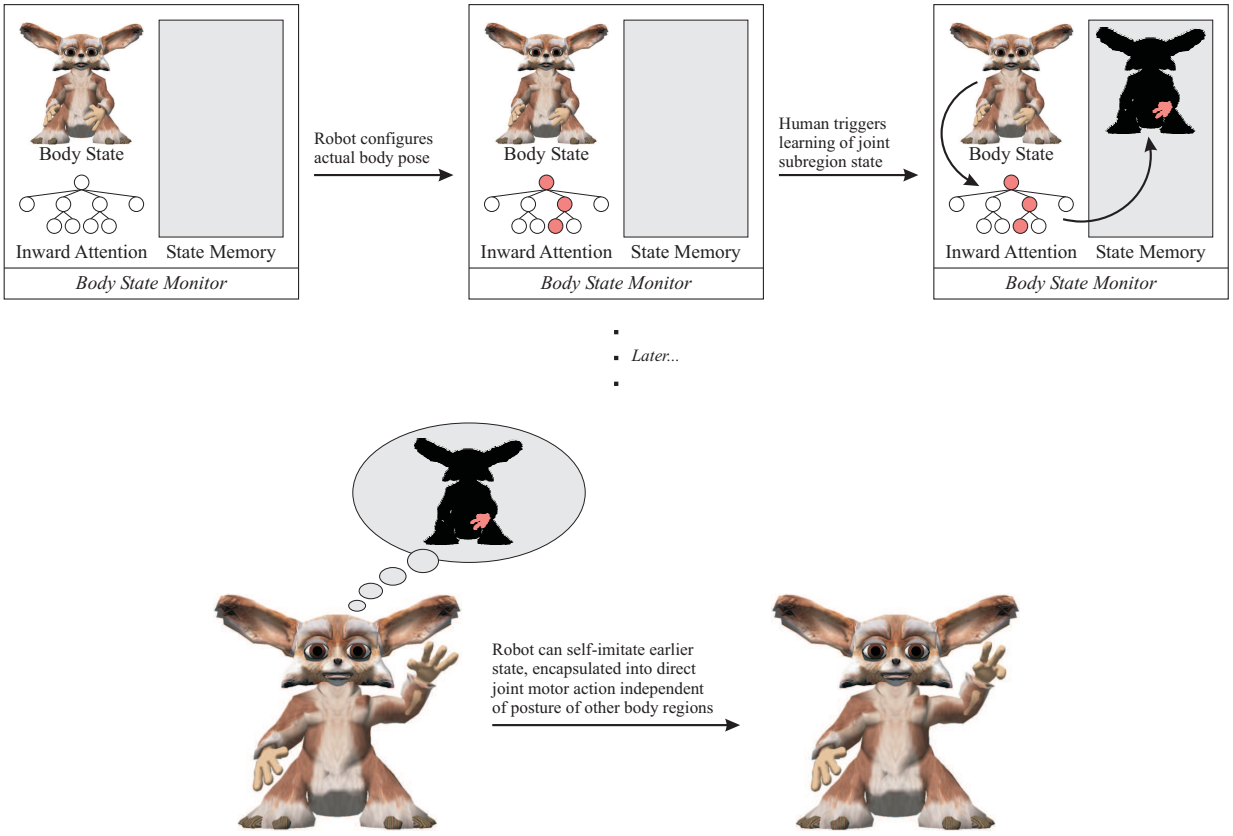


Figure 3-5: Self-imitation of an example body state.

The modular nature of the architecture means that behavioral control components handling perceptions, beliefs, actions, motions, and so on, are attached to the model as needed. For bodily self-modeling, the principal components of interest are of course the action and motor systems. An overview of the specific functioning of the action system in our version of the architecture, `C5M`, is given in 7.2.1, and the motor system is similarly described in 7.3.1. However, the specific implementation details of these systems are not important in this discussion. What is relevant is the consequence of this modularity: separate instances of the identical action and motor systems can be used to control separate instances of the kinematic model, without requiring explicit modification of any component.

It is therefore possible to give a robot *two* identical kinematic models; one of which ultimately controls the actual robot, and the other of which can be used for self-simulation. Because the behavioral components are also identical, the robot can be as confident as it is conceivable to be that the stimuli applied to these components will produce the same results on the mental simulator as on the robot itself, because the observable state mechanisms are exactly the same. The only difference is that one controls the real robot and one does not. This latter "imaginary robot" can therefore be used to make accurate inferences and predictions, as is desired.

Naturally, in practice there are a handful of caveats to this. Occasionally, modules do need to know whether they are executing on the real or imaginary robot. The most important example of these is behaviors which can themselves trigger simulation behavior. To avoid infinite recursion,

these specific parts of those behaviors must be suppressed. The other example that has been encountered is a behavior that contends for a resource that is not part of the kinematic body model, such as the audio speaker (e.g. during lip-synchronized speech). Again, requests for use of these resources needs to be protected by a check to confirm execution on the real model.

From a philosophical standpoint, this approach can be seen as a hybrid between the modular mental simulation approach espoused by psychologists such as Barsalou (Section 2.1), since existing modules are largely reused without alteration, and a representationalist approach, since the simulation ultimately occurs on a completely decoupled internal body representation. In answer to any antirepresentationalist criticism, however, I would point out that as described here the process is purely a physical simulation; the imaginary robot does not have perception and belief modules attached. Most of the critiques of representationalism concern the necessity for additional structures to interpret the internal representations of perceptions and beliefs. In this case, once these modules are also reused, the relative contribution of the additional explicit physical representation will be substantially decreased, and the overall effort will be largely akin to the psychological self-simulation theories described earlier.

For now, the mental self-model is concerned with performing two primary introspective functions for the robot. The first is to allow the robot to model itself behaving as itself — that is, generating self-motivated action. The second is to allow the robot to use its common ground with other agents to model itself behaving as those others. These are described in the following two sections.

3.3.1 Self-Modeling Motion

Given the arrangement of duplicate kinematic model and behavioral modules described above, the mechanism by which the robot models its own motion is straightforward. Actions are triggered, and motions generated, within the components assigned to the robot's imaginary doppelganger, and the results observed by simply observing the kinematic output. Before proceeding to the nature of this observation, it is necessary to examine the practical reason for doing it. This amounts to a general method for converting known actions — which may exist in the robot's repertoire in any of a number of elemental forms, from joint-angle series to interpolated blends — into arbitrary time series representations. During mental simulation, the robot can select an appropriate representation in order to make use of particular analytical algorithms.

The first such representation is for the contemplation of time itself. It is not hard to conjure up reasons why the robot might benefit from being able to estimate in advance how long a particular action might take, or to know what the state of its body will be at a particular time after commencing the action. An example might be deciding whether or not to assist a partner with a task, by attacking a subtask. If the robot knows it can attend to the subtask before the partner is likely to attempt it themselves, it can do so in confidence that it will not be getting in the way; otherwise it might do better to wait and observe the next action on the part of the partner before making a decision.

At present, the mental self-model is used only for determining how long actions are likely to take, both *in vacuo* and in real-world situations involving other actions in a known sequence. The central representational issue here is, of course, what should be used as the basic unit of time. The passage of time in the real world is measured in seconds, and the elapsed time thus varies according

to the speed with which an action is performed. However, the robot has access to a deterministic time representation in terms of the "ticks" that represent cycles of its internal processing system. All other things being equal, an action based on a pre-animated motion will take the same number of ticks to perform on a subsequent occasion as it did on the first. A tick-based representation has a pertinent advantage for mental simulation as it enables the imaginary robot to be clocked at a different rate from the real robot, enabling tractably fast predictions.

Durations of actions that are timed in ticks can, of course, be converted into seconds if the speed of the real robot's clock is known. The historical speed of the clock is easy to calculate, but the clock speed during actual execution of an action may be impossible to predict with absolute certainty if the speed fluctuates with processing requirements. Nevertheless, the utility of a tick-based representation is such that it has been selected, and the robot's operating clock speed is currently predicted using a historical average over a short window.

The second class of representations that is considered is the collection of possible time series of vector features describing aspects of the robot's body position. Many of a robot's motions may already be stored in an explicit series representation, such as the joint-angle series used for Leonardo's pre-designed animations (Section 7.3.1). However, it may be more advantageous to be able to express them in terms of different feature vectors. For example, the spatial coördinates of the end effector position are popular for some applications, and can be expressed in a variety of frames of reference. Similarly, some robotics researchers have preferred to attempt to reflect the overall envelope of the motion, rather than the fine details of joint angles and end effector positions, using features such as the kinematic centroid[139].

I therefore chose to implement a generic motion analysis framework for the robot's mental simulator that allowed time series data to be collected, analyzed, and synthesized, independent of the underlying feature set. A popular technique for supporting the comparison and classification of time series data is Dynamic Time Warping (DTW). The DTW algorithm is used to find the optimum alignment between a pair of time series by non-linearly manipulating the time axis of one of them. Essentially, one series is appropriately "stretched" and "squeezed" at various points in time along its length such that it best matches the other. The aligned series are then more appropriately prepared for operations such as matching, similarity analysis, and the extraction of corresponding regions.

Dynamic programming techniques can reduce the DTW algorithm to quadratic complexity, which is still prohibitive for real-time analysis of long motion sequences observed at a high sample rate. However, accurate multiresolution approximation algorithms exist that achieve linear complexity with subpercentage error rates in many cases [255].

The DTW technique would allow the robot to account for any temporal discrepancies between motions observed on its imaginary self-simulator and those observed on its real body. Furthermore, it could perform one-to-one comparisons between motions that had been represented in this way. Yet a more powerful technique might allow the robot to perform one-to-many motion comparisons. In other words, to find a one-to-one match if it exists, but otherwise to decompose a motion into a combination of other motions that might approximate it. For this, the time series must be warped in space as well as time.

It would be convenient if, given a time series representation of a motion $\mathbf{x}(t)$, it would be possible to represent this as a combination of some set of prototypical actions in the robot's repertoire, $\mathbf{x}_p(t)$.

It would be even more convenient if this could be a linear combination, proportioned with weights w_p according to the principles of linear superposition:

$$\mathbf{x}(t) = \sum_{p=1}^{P} w_p \mathbf{x}_p(t)$$

When an action was observed on either the real or imaginary robot, it could be matched against the motor repertoire by solving the above equation for the weights w_p that would have produced it. This information could then be used by the robot to make a variety of predictions, that might concern aspects of the nature of the action itself, or what other known action or combination of known actions might be an appropriate substitute if necessary.

However, this would require the ability to simultaneously align the trajectories in both time and feature space. For a given pair of multi-dimensional sequences $\mathbf{x}_1(t)$ and $\mathbf{x}_2(t)$, the correspondence problem is ill-posed: there is an infinite number of potential pairs of spatial and temporal displacements $\xi(t)$ and $\tau(t)$ that could "map" one sequence onto another:

$$\mathbf{x}_1(t) = \mathbf{x}_2(t + \tau(t)) + \xi(t)$$

This precise problem was tackled by Giese and Poggio in their work on Morphable Models[105]. I have therefore implemented a system based on the algorithms they developed. The spatial and temporal displacements are represented as deviations from a reference sequence $\mathbf{x}_0(t)$. The system is constrained to a unique solution by balancing the importance of spatial versus temporal accuracy with the application of a linear gain λ:

$$\mathbf{x}(t) = \mathbf{x}_0 \left(t + \sum_{p=1}^{P} \tau_p(t) \right) + \sum_{p=1}^{P} w_p \xi(t)$$

$$[\xi, \tau] = \text{argmin} \left(\int [|\xi(t)|^2 + \lambda \tau(t)^2] \mathrm{d}t \right)$$

This supports both one-to-one and one-to-many comparisons, through selection of the reference sequence $\mathbf{x}_0(t)$. To directly compare two sequences against each other, one of them is simply made the reference sequence, and the residual spatial and temporal errors can be examined when it is warped onto the other using the technique in [105]. To compare a sequence against a prototype set, the reference sequence is computed as a series that averages the feature values of each of the prototype sequences. The set of linear weights for these prototype trajectories are then determined by solving the following linear parameter estimation problem:

$$\mathbf{w} = \text{argmin} \left(\int \left[\left| \xi(t) - \sum_{p=1}^{P} w_p \xi_p(t) \right|^2 + \lambda \left(\tau(t) - \sum_{p=1}^{P} w_p \tau_p(t) \right)^2 \right] \mathrm{d}t \right)$$

This linear parameter estimation problem can be solved using conventional linear least squares. However, Giese and Poggio note that as the size of the basis set becomes large, linear least squares

exhibits poor generalization properties. Instead, they showed in [105] that a better method to use in this case is one in which the capacity of the approximating linear set is minimized, and proposed the Modified Structural Risk Minimization algorithm to enable the use of *a priori* information as a constraint. In this technique a positive weight matrix is used to punish potential linear combinations of incompatible basis actions.

This approach suits the application to communicative human-robot interaction, because the robot often has contextual knowledge of the types of actions that are likely to be appropriate in a given situation. For example, the action of pressing a button looks extremely similar to that of patting a stuffed animal, and the linear decomposition of an example reaching action could appear to be an indiscernible mixture of the two if all basis actions were allowed to be weighted equally. But by using the object referent (a button versus a stuffed toy) to select an appropriate weight matrix (that could be automatically generated from past experience), the robot would be able to select a much more confident match.

There are several limitations to this approach. One is the necessity for the balance between spatial and temporal warping of the prototype sequences (the parameter λ) to be decided based on *a priori* information. Another is the question of whether or not linear superposition of the prototype actions is appropriate in the first place. Giese and Poggio showed that motion applications exist in which the pattern space manifold is sufficiently convex that many potential weight combinations can produce recognizable results. However, for which kinds of motion classes this is true, and how this depends on the original feature selection, is not known.

Since the ability to perform real-time action matching is a small component of this thesis, I have not made comprehensive attempts to determine the answers to these questions. Instead, since the constraints on the selection of prototype actions provided by the communicative interactions of interest are particularly strong, I have generally used this technique to perform one-to-one comparisons or linear decompositions of small basis sets comprised of largely dissimilar sequences (see the following section).

Nevertheless, the principles behind creating this general framework for time series comparisons are independent of whether or not the linearity assumption holds. The spacetime alignment interface allows the robot to prepare self-simulated motions for subsequent analysis, while retaining flexibility as to the choice of which features will be used to provide the underlying data for that analysis and which algorithms, whether ultimately linear or non-linear, will be used to perform them.

3.3.2 Self-Modeling Imitation

The use of the mental self-model to represent physical imitation behavior of a human demonstrator is a similarly straightforward mechanism. Since the robot now has bodily common ground with the human via its learned body mapping (Section 2.2). It is therefore able to directly map the human's motions onto the body of the imaginary robot, via a perceptual module that continuously accepts motion tracking data and uses the body mapping to convert it into the robot's joint space. In essence, the robot is "imagining" itself doing what the human is doing. This process occurs whenever the imaginary robot is not occupied with the specific motion simulation described in the previous section.

One primary function of doing this is to drive certain aspects of the robot's inward self-attention system, described below in Section 3.4. A broader motivation, however, is to enable self-modeling to act as the mental link between the low level "mirror neurons" of the body mapping, and the high level process of actually interpreting and executing the imitation. This is the part of the robot's imitation subsystem in which demonstrations are actually mapped into its motor space.

An important reason that such a link should be treated separately from the underlying body mapping is that the robot needs some way of segmenting imitative demonstrations and pruning them based on whether or not it is physically possible for the robot to actually learn them. If the robot were to simply map the demonstration directly into its physically embodied motor space as it occurred — in other words, to simultaneously mimic the human's motion — the robot would face difficult decisions at transition points where the demonstration began to exceed the robot's physical capabilities. Continuing with the mimicry using some subset of the demonstration could produce unpredictable and potentially damaging results, whereas simply abandoning the demonstration would be frustrating for the human teacher. Processing the demonstration "off line", in the robot's mental simulator, allows it to determine potential problems and how to deal with them in advance, before constructing a response to the demonstrator.

One example of this is the simple exceeding of joint limitations. The human body is much more flexible and versatile in some respects than any humanoid robot, and the body mapping mechanism allows the robot to determine when the human is adopting a posture that would violate the robot's kinematic model. When this is discovered during mental simulation, the robot can continue to model the imitation with limitations placed on the mapping that prevent these boundary violations. If the resulting motion sequence does not violate other aspects of the model — such as segment collisions — the robot can then choose to execute the modified action as a trial, in order to seek feedback from the human. If the modified sequence does have problems, the robot can simply refuse to perform the action, or use other communicative mechanisms to indicate that the human should repeat it or try something else.

Another main reason to treat this imaginary mapping process as an autonomously self-aware module is to support the imitative recognition of actions that the robot already knows how to do. If the human wishes to communicate that the robot should do a particular action, the most intuitive means of doing so is for the human to simply perform the desired action as a demonstration. The robot needs to map the human action onto its own body and then compare it with its repertoire. Rather than decide when to start mapping the human, record the motion, stop mapping, and perform the comparison, it makes more sense for the robot to already be engaged in mentally imitating the human, and then simply compute the comparison based on some short term memory of the mental self-model.

This means of comparison eases the requirement for precisely selecting the start and finish points of the demonstration. Assuming a long enough memory buffer, the robot can respond to an inaccurate determination of the start point by simply moving the buffer read point and trying again. Similarly, if the robot is not certain of the end point, if sufficient processing resources are available it can spawn regular recognition processes at various points near the likely end of the demonstration, and then select the one that produces the most likely match.

As stated above, the robot therefore continuously processes motion tracking data, kinematically decomposes it as described in Section 2.2, and applies it to its imaginary body model whenever the model is not otherwise in use for motion self-modeling. Of course, there is nothing to prevent the

robot from having a *third* self-model specifically for imitation, but philosophically speaking there is little to support a further proliferation of internal models. Should the robot then spawn a new imitative self-model for each human it can simultaneously perceive? Arguments could certainly be raised in favor of this arrangement giving it the most flexible capabilities. However it is more efficient, and more "natural", for the robot to instead multiplex its imagination by just attending to the mental task at hand.

The imitative self-modeled motion data is then used, when necessary, to compare the human's actions (as mapped into the robot's body space) to actions in the robot's repertoire using the dynamic warping technique outlined in the previous section. For example, the results of one such comparison are shown graphically in Figure 3-6.

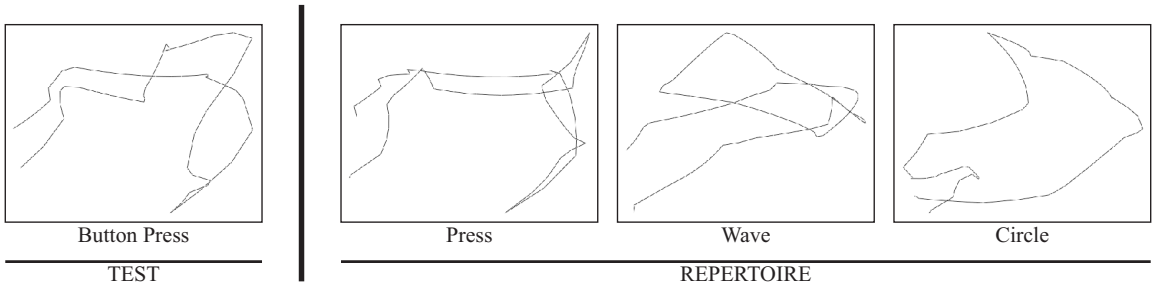


Figure 3-6: Example comparison of a novel button press arm motion against a repertoire of existing arm motions, including a known button press. The graph tracks represent the positions traced out by the robot's wrist coördinates over time.

In this comparison, a novel button press motion was compared with an action repertoire consisting of another button press motion, a hand wave motion and a motion in which the hand was used to trace out a circle. The graphs were generated by automatically collecting the 3-D position of the robot's wrist over time, rotating these points by forty-five degrees and then projecting them onto the xz-coördinate plane. The system recognized the novel button press to be composed primarily (86%) of the known press motion (Table 3.1), indicating to the robot via a threshold function that the motion was most likely to have been a type of button press.

Action	**Composition**
Press	0.86471
Wave	0.10593
Circle	0.02893

Table 3.1: Numeric results from the example comparison illustrated in Figure 3-6 using the morphable models technique. The decomposition shows the novel button press to have much more of the character of the press prototype motion than that of the wave or circle prototypes.

As a result, if the human was to have demonstrated this motion, the robot would be able to theorize that the demonstration was a form of button press action, since that was the closest match. If the human was demonstrating the motion in order to convince the robot to perform it, the robot could then do so. Alternatively, if the human was performing the demonstration with the intention that the robot should imitatively learn to refine its own action according to some difference between

the human's method of performing it and that of the robot, the robot would then be in a position to further analyze the deviations between the two sequences and attempt to determine the salient features that the human might be trying to teach.

3.4 Inward Self-Attention

It has become reasonably well accepted that an attention system is a way of managing a robot's perceptual capabilities that confers certain advantages. A detailed justification can be found in [46], and I will only briefly summarize. In general, the main benefits an attention system provides are dual approaches to the issue of extracting truly salient factual information from the environment. Allowing the robot to focus its attention contributes to a narrowing of the potential search space in any particular situation by offering a reliable means of discarding extraneous perceptual data; conversely, when the robot's sensors are not omniscient (as is always the case in the real world), it becomes a mechanism for increasing the data flow in promising areas by enabling the robot to bring its powerful directed perceptual tools to bear upon them.

Despite these advantages for attention systems that essentially operate in the space between the robot's sensors and the environment, I am not aware of any research project that has specifically addressed the direction of attention within the robot's own body, inside the "skin" of its environmental sensors. Perceptual routines that merely access internal state variables do not fulfil this description, much as simply keeping track of external state does not constitute an attention system, as the nature of such a system implies a means of mediating across all of the state variables used by the various perceptual mechanisms.

An introspective look at human bodily attention provides anecdotal justification for investigating this concept. Our attention to aspects of our own bodies can frequently be traced to the signals from environmental sensors such as touch and vision. However there are certainly occasions in which we become acutely aware of our bodily configuration and state independent of sensing of the external environment; examples include attending to our own posture, unconscious habits, and internal pain. Coupled with the advantages of improved disambiguation skills and analytical data collection that follow similarly as they do in the external case, it was considered worthwhile to implement an inward self-attention mechanism for Leonardo.

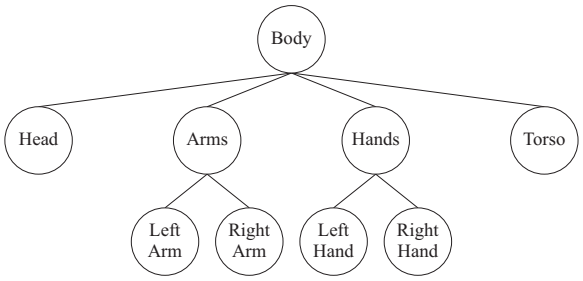


Figure 3-7: Leonardo's bodily attention region hierarchy.

Leo's inward attention system is based on breaking its body down into hierarchical regions that correspond as closely as possible to functional communication units (Figure 3-7). The root of the hierarchy is, of course, the robot's entire body. This is then subdivided into torso (postural

communication), head (facial expressions and speech), arms (body language and gesture), and hands (emblematic gestures). Body regions exhibiting resource symmetry are then further broken down into their left and right side counterparts. A graphical representation of Leonardo showing the various body regions is shown in Figure 3-8.

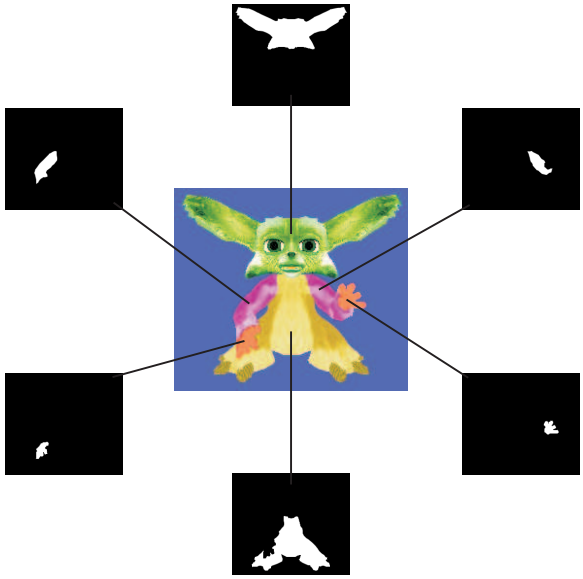


Figure 3-8: Leonardo's bodily attention regions displayed as they concern the actual body of the robot.

Leonardo's behavior engine originally contained a minimal external attentional system which used an action group (see Section 7.2.1) to direct the animated head and eye pose towards objects of attention. Likewise, the inward attention system has been placed within the action subsystem of the robot, due to its tight coupling with actions. However it is made visible to all other subsystems that may wish to query it to determine which body region, if any, is a candidate for being attended to. This determination is made according to the activation level of the regions, and the specificity of the activation. How these regions are conceptually activated and habituated, and the implementation details of this process, are described in the following sections.

3.4.1 Activation and Habituation Schemes

Each of the different body regions have somewhat different activation and habituation schemes, in reflection of the different functions that each tends to perform during human-robot communication. In addition, however, there are general activation and habituation effects that apply to all regions.

Perhaps most straightforward is the behavior of the hand regions. Since the hands are currently the only part of Leonardo's body that is equipped with skin sensors, the primary unique mode of activation is through touch. However, simply activating the inward hand attention when the skin is touched is more appropriate to the behavior of an external attention system rather than an internal one. Instead, the activation of the inward attention concerning the hands is tied to the touch reflex actions described in Section 3.2.

Recall that the primary communicative application in this region is for the human to be able to draw the robot's attention to its hand for the purposes of self-directed stimulus enhancement. It is

therefore undesirable for the robot to draw attention to its hand configuration every time it contacts an object. At present, Leonardo does not manipulate objects, but using the reflex mechanism to stimulate inward hand attention anticipates this behavior, as additional context (type of touch, hand-eye coördination and spatial short term memory) can easily be used to adjust the reflex actions so that they do not fire when Leonardo plays with objects, preserving the nature of the robot's inward hand attention.

The reflex actions are designed to incorporate the kind of attentive response that could be expected if the sensors were driving attention directly. Different reflexes activate attention at different rates — the response to stroking the palm to trigger a grasp reflex is different to pinching a finger to explicitly toggle its state. Furthermore, in addition to changing the hand configuration the reflexes continue to fire for a short period to allow a range of activation responses depending on how the human might be interacting with the hand. For example, if the human simply touched the robot's hand to provoke a response, this may be less important than when the human is deliberately molding the robot's hand in a more drawn-out process, and the robot should clearly pay more attention to the latter.

Habituation occurs according to the time since a change in the interaction with the robot's hand. This occurs in two conceptual areas. First, the reflex actions are set to stop triggering after the initial firing period unless the nature of the low-level stimulus (the thresholded FSR values) changes. This stops further activation of the region, allowing the time-based habituation to take over. The time-based habituation runs continuously, sending linear inhibition signals to the attention function (Section 3.4.2) for all regions.

The robot's arm attention region is treated somewhat differently, as it has no skin sensors, but is still intended to be activated by aspects of the communicative interaction between the human and the robot. In particular, since the robot is mapping the human's body onto its own body as part of its "mirror neuron" approach (Section 2.2), the human should be able to draw the robot's attention to its arm posture through drawing attention to his or her own. The inward arm attention therefore draws from aspects of the real-time human motion tracking data.

First, an acknowledgement is needed that the motion tracking data can be unreliable, even to the point of seeming to be able to extract a human pose even when no human is present in the scene. Therefore the kinematic decomposition of the human's pose contains warning flags that are set when the motion tracked pose appears to be one that is inappropriate for the robot to even consider. If these warning flags are set, no activation of the arm region based on the "mental modeling" of the human is allowed to take place (inward attention can still be activated by the robot's own actions, of course, described later in this section).

Waving the hand is already used as a trigger for the minimal external attention system, directing Leonardo's on-board vision via eye and head pose. However, if the human is trying to teach the robot an arm configuration, he or she is likely to be holding the arm in question motionless for the robot to analyze, rather than shaking it about to attract attention at the same time. Nevertheless, the human usually has two arms, and may well be using the other hand to deictically gesticulate towards the demonstrative arm. Leo's arm attention is thus communicatively activated in situations when one arm is mostly stationary (according to the mean squared joint velocity) and the other hand is moving and located close to the first arm. Once again, this activation is a measure of novelty, and will stop firing if the situation does not change, allowing timed habituation to become dominant.

The inward head attention region is similarly activated with the use of the hands as deictic communicators. The robot has the ability to visually track facial features, and this has been used to support imitative facial mapping construction (described in [44]); but for full facial imitation to occur as part of a generalized interaction, the human needs to be able to draw the robot's attention inward to its own face as it imitates the human's. The robot's minimal external attention is already predisposed to fixate on human faces unless other things are more "attention grabbing", so convincing the robot to attend to the human's face in order to generate the tracking data is already dealt with. The robot now needs to be made to understand that the true locus of attention is its own mapping.

This is accomplished by taking account of the position of the human's hands. If the hands are close to the face, "framing" or otherwise indicating it, this is considered a cue to inward facial attention. In a real human-human interaction, the inward attentional shift might also be communicated by a combination of exaggerated facial expression, proximity of the teacher's face, and of course the context of the interaction. Our sensing of whether or not the expression is exaggerated is too crude, so we instead concentrate on the presence of such a framing gesture as a sufficient indicator. In the human-human case, depending on the nature of the interaction it might be sufficient but not necessary, and so this is considered good enough for the present.

Finally, specific activation of the inward torso attention region is considered. This region is more difficult to deliberately draw the robot's inward attention to, due to technological constraints, so it is currently not implemented though has been planned for when the technology is available. Consider a human communicating a posture to another human for imitative transfer. It is likely that the teacher would simply demonstrate the posture, possibly somewhat exaggerated, and the context of the interaction would be enough for the learner to copy it. The teacher almost certainly would not use the hands to indicate the postural elements, nor use proxemics as a marker. I therefore propose that large changes in the spine curvature during a teaching interaction would be the most appropriate trigger for the robot's inward torso attention.

However, the Vicon motion tracker being used is currently set up to model the human with a simplified skeleton that maximizes the accuracy of the head and arm joints, rather than spine curvature. This is primarily due to practical difficulties with modeling the spine — the more articulated the spine is made, the more tracking point markers are needed, and the greater the inaccuracy caused when these markers inevitably move due to the elasticity of the motion capture suit. To improve the ease of interacting with the robot, and because there are other ways to influence the robot's torso configuration (such as speech), this feature of the inward attention system remains for future implementation.

As mentioned earlier, a continuous timed habituation across all regions constantly delivers a linear inhibition that dominates when simultaneous activation is not occurring. All regions are further affected by one further generic pair of activation and habituation mechanisms.

The purpose of inward attention is for the robot to be able to include the activity of its own body as part of the context of the ongoing communication. This activity must include a recent history, because the human observing the robot will certainly be aware of this history and include it as part of his or her communicative context. The actions that the robot performs thus activate the respective inward attention regions of the body parts that they control. This enables the human to use deictic phrase fragments such as "that hand" to refer to the hand that just performed an action, and have the robot have a good chance to correctly infer what is meant.

However, the robot also performs many actions that are not instrumental to the communication, and if they are allowed to continually activate the attention regions (e.g. a full body idle motion that involves all of the robot's body regions), they will saturate the legitimate inward attention signals. Therefore this activation again occurs at the action level. Actions that are performed frequently, in particular looped idle motions such as "breathing" and blinking, have their inward attentional activation signals suppressed. This is clearly not the same as habituation of the regions themselves, as this would also swamp the activation signals of legitimate actions, but can instead be thought of as an action-level habituation.

3.4.2 Bodily Attention Implementation Details

The implementation of the inward bodily attention system on Leonardo is straightforward, and concerns two main details. First, the robot constructs its own attention region map based on the actions in its innate repertoire, and second, the attention system maps linear activation and habituation inputs onto a non-linear output function.

In an idealized, purely developmental scenario, a robot would enter the world equipped with minimal innate knowledge of the joints, segments and actuators which make up its physical embodiment. It would then proceed to discover these through exploratory interactions with the physical environment and communicative interactions with other agents such as humans. Clearly, I am not taking this approach, as I have laid out above the division of the humanoid robot's body into the various functional subregions and the scenarios by which these are activated and habituated. Nevertheless, I have designed the inward attention system to support a more developmental approach in the future. These functional subdivisions are not encoded within the inward attention itself; rather, they are encoded within the actions and the action system that are used to construct the inward attention system when the robot is started up.

When the inward attention system is initialized, it is empty, containing only the root node for the robot's whole body. The population of the inward attention system, and thus the subdivision of the robot's body into attentive regions, is subordinated to the action repertoire structure that is similarly populated with actions as the robot starts up (Section 7.4.3.2). The discussion of these two structures is separated, because the inward attention system is not dependent on the action repertoire *per se* — it is merely a convenient way for the robot to compress a developmental upbringing into a quick traversal of every action it knows how to do. If a developmental "upbringing" was desired, these structures could be decoupled, and the inward attention system populated via exploration rather than an exhaustive search of innate capabilities.

Leaving a complete discussion of the action repertoire for later, suffice it to say that the actions with which it is populated are tagged with region labels that indicate to which parts of the body each action applies. These tags are thus simultaneously used to populate the inward attention mechanism. Thus once the robot "knows" how to do everything it can do (within generalized boundaries), it also knows the generalized functional structure of its own body.

Each body region, once added to the inward attention system, is assigned a standard normalized positive activation value that can range between 0.0 and 1.0. This output range is what any subsystem querying the inward attention system will receive. However an important implementation detail is what maps the input to the inward attention system, in the form of discrete activation and habituation signals, with the ultimate output value. From a modularity standpoint, it is convenient

for the incoming signals to be essentially linear. For example, an internal stimulus that is twice as "strong" (in some appropriate sense of the term) as another internal stimulus should be permitted to send an activation signal that is twice as large; and the time-based continuous habituation should be able to send the same strength of habituation signal regardless of what else is going on at the time. This simplifies the generation of activation and habituation signals by allowing their sources to have control of their own signal levels.

However, at the same time there are advantages to having the inward attention system itself exhibit non-linear behavior, such as smooth rise and fall of the output levels in response to discontinuous discrete input. This can of course be accomplished by filtering the input values, but the ideal behavior of the attention system can be more intuitively explained in terms of the shape of the desired output mapping. Any given region should require more activation to initially start having an attentive effect than to increase that effect in the continued presence of the stimulus; conversely, a region should have a smooth attentive peak that responds slowly to both further stimuli and initial habituation, but which then ramps down quickly when the need for attention is truly past.

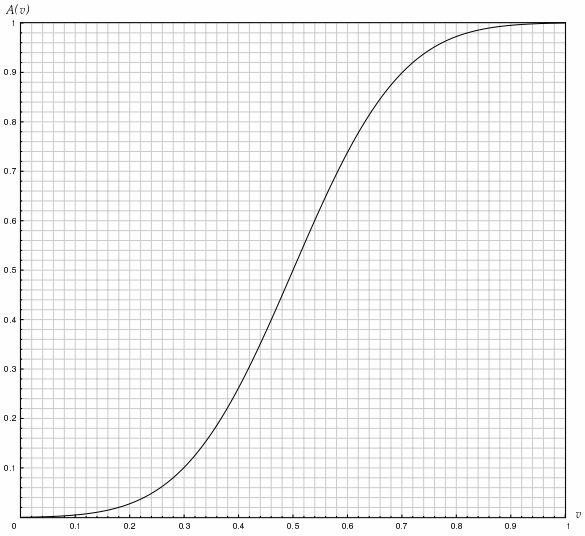


Figure 3-9: The inward attention activation output function.

In other words, the output function should have the appearance of a standard psychometric curve. The characteristic behavior of such a function for attentive purposes is to resist noise at the extremes (inattentive or highly attentive states), while making it easy to set linear thresholds in the middle section. A classic function that has this particular shape is the Gauss error function, denoted erf, defined as:

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} \mathrm{d}t$$

This function is odd, with a domain of $(-\infty, +\infty)$ and a range of $(-1, 1)$, whereas the attentional curve needs to have both a domain and range of $[0, 1]$. The range can be achieved by scaling and shifting, but due to the original infinite domain the function must also be truncated. The

derived inward attention output function A, for a given activation value v, that accomplishes this transformation and preserves the closed domain and range endpoints is therefore given by:

$$A(v) = \frac{1}{2}\left[1 + \frac{\text{erf}(\sigma(v - \frac{1}{2}))}{\text{erf}(\frac{\sigma}{2})}\right]$$

The parameter σ defines the width of the truncation window imposed on the Gauss error function, and thus the steepness of the activation and habituation curves at the attentive extremes. In practice a value of $\sigma = 4.5$ was chosen. A plot of the inward attention output curve under these conditions is shown in Figure 3-9.

This implementation concludes the set of self-awareness support structures that I have designed to work towards the ultimate goal of enabling general, open-ended human-robot communication. The following chapters proceed to examine a range of specific classes of human-robot communication that require coördination. Inward attention and self-awareness support in general are not necessary for all of them, but are part of the suite of tools that can be drawn on to facilitate that coördination.

Chapter 4

Minimal Communication

It is common in the youthful field of human-robot interaction to take a top-down approach to expression and communication, by modeling interactions on well known aspects of high level human-human communication. Examples of this are manifold: anthropomorphic robot designs, human-specific affect, the use of natural language, to name a few. Even though the underlying architectures may not necessarily represent strict top-down hierarchies, as in the case of behavior- or software agent-based systems (e.g. [101, 215, 235]), the means by which the robots achieve common ground with humans have been designed with complex human response in mind. The justification for this is perhaps best summarized by this quote from Peters et al.:

> Viewed in the light of situatedness and embodiment, ... scaffolding theories suggest that for a robot to interact with a person in a human-oriented environment, as an assistant or to solve problems, the robot should be anthropomorphic in at least some physical and behavioral aspects. [235]

I do not disagree with this. However, I believe that there are also good reasons for exploring a baseline level of communicative support for human interaction that sits somewhat below the level necessary to act as an assistant or adjunct problem-solver. I refer to this level or layer as "minimal communication": the most fundamental perceptual and expressive skills of a robot that allow it to be perceived as convincingly interactive in its exchanging of state information with a copresent human. In other words, if these skills are lacking then a human directly engaging the robot is likely to receive the impression that it is somehow "not quite there"; perhaps on the lines of a kind of "Turing test" for interactivity.

The key aspects of this definition are thus convincing interactivity and physical copresence: bodies are important. Other notable features are what is not mentioned: the interaction may not have an *a priori* known purpose (e.g. a task), and communication behaviors at this level may not necessarily be readily anthropomorphized.

Due to the understandable focus on supporting communication for higher level tasks, minimal communication for the express purpose of natural and organic behavioral appearance and convincing interactivity has not been widely investigated in the human-robot interaction community. In this chapter I will first expand on the nature of minimal communication, and then the reasons which

justify its treatment as a distinct area of consideration. I will them present a novel application of minimal communication to human-robot interaction, followed by details of the implementation and deployment of a robotic installation that demonstrated these capabilities.

4.1 The Nature of Minimal Communication

Minimal communication has been loosely defined above in terms of a baseline condition that is itself not strictly constrained. It is put in these terms in part to avoid the danger of being bogged down in reductive semantics over what constitutes communication.[1] For example, it could be argued that simple human avoidance on the part of a mobile robot is a communicative behavior, as it 'communicates' a 'desire' on the part of the robot not to crash into people. Rather than attempt to refute each such example, the loose definition merely states that in order to be considered minimally communicative, a human must be convinced that such a robot is genuinely interactive. Said human's specific preconditions might be that the robot reacts differently towards him or her than towards an inanimate object, exhibits coherent responses to his or her activities, and displays directed behaviors that convey aspects of its internal state that are critical to a natural shared interaction. In some cases behavior along the lines of the aforementioned exemplar might qualify, and in others not.

Another perspective of minimal human-robot communication can be obtained with reference to the wider Human-Computer Interaction literature. In particular, although for this purpose we must take care not to assume the foreknowledge of a clearly defined task, human-robot interaction in general perhaps best comes under the purview within HCI of Computer-Supported Coöperative Work (CSCW). CSCW is defined as computer-assisted coördinated activity carried out by groups of collaborating individuals[114], and in this case we consider that said computer support manifests itself in the fact that one or more of the collaborating individuals are autonomous robots. The concept of minimal communication can thus be expressed as a minimal exchange mechanism for what the CSCW literature refers to as "awareness" information: the information that participants have about one another.

Unfortunately, while being instructive as to the types of communication that might be part of minimal awareness, the CSCW literature has not yet provided a standard definition of awareness. Of the many definitions of types of awareness that have been proposed, the following selection (based on a larger set presented in [92]) shows the most relevance to minimal communication for HRI:

awareness An understanding of the activities of others, which provides a context for your own activities. [89]

conversational awareness Who is communicating with whom. [290]

informal awareness The general sense of who is around and what others are up to. [116]

peripheral awareness Showing people's location in the global context. [117]

[1] An analogous example is the perennial fruitless debate over what constitutes a robot. A stereotypical line of argument might proceed as follows: the definition of a robot is a machine with sensors and actuators; an elevator has sensors (buttons) and actuators (doors and winch); robots are thus already commonplace in urban societies.

social awareness Information about the presence and activities of people in a shared environment. [242]

proximal awareness The more an object is within your focus, the more aware you are of it; the more an object is within your nimbus, the more aware it is of you.[2] [20]

Taken in the absence of a specific shared activity, these forms of awareness suggest the following general communication components that a robot should support in order to appear minimally aware:

- The ability to monitor the presence or absence, means and targets of messages generated by others, including the intention to communicate;
- The ability to direct status messages generated by oneself to appropriate destinations, including the intention to communicate;
- The ability to recognize and communicate loci of attention and spatial presence, including personal space.

In response to the definitions of awareness from the CSCW literature from which the above were sourced, Drury, Scholtz and Yanco proposed a specific definition for HRI awareness:

> HRI awareness (base case): Given one human and one robot working on a task together, HRI awareness is the understanding that the human has of the location, activities, status and surroundings of the robot, and the knowledge that the robot has of the human's commands necessary to direct its activities and the constraints under which it must operate. [91]

Reducing the task to the minimal case of a convincing interaction, I propose the following definition for minimal HRI awareness that satisfies the three component conditions specified above:

> Given one human and one robot interacting, minimal HRI awareness is the understanding that the human has of the location, spatial presence and status of the robot, and its responses and responsiveness to communications directed towards it, and the knowledge that the robot has of the human's attempts to affect its behavior through communication and action.

Two aspects of the minimal communication that underlies this minimal awareness are not obvious from the above definition and must be specified here. First, a true minimal communication capability excludes all formal symbolic language. Some of the research efforts related to minimal communication have looked at the minimum spanning language templates that permit task-oriented communication, e.g. [313]. However since convincing interactions can take place without the use of natural language, it is not part of the minimal set.

Second, there is no requirement for an anthropomorphic design on the part of the robot. Minimal communication must be able to support the exchange of basic information between a human

[2] The authors did not provide a name for this type of awareness; the term 'proximal awareness' is mine.

and a robot with no humanoid physical features. One way of looking at minimal communication is as the communication channel between two species (i.e. humans and robots) that may be morphologically separate but are able to recognize one another. For example, consider human-dog interactions: certain communicative expressions are readily anthropomorphized, such as using eye contact to communicate dominance and threat response; whereas others, such as piloerection of the hackles to indicate aggression, are not, but may be easily recognized once familiar[219, 93, 269]. What about highly abstracted, nonanthropomorphic behaviors such as the "dances" bees use to communicate spatial information to one another, but which a human who is not an expert apiologist would certainly be unable to interpret (e.g. [197])? This ability alone does not satisfy the three criteria above for maintaining minimal awareness, which is why bees themselves are not generally considered particularly interactive, but one can certainly imagine ways in which a bee-like robot might be able to interactively share basic information with a human, such as navigational assistance or direction of attention to an external target.[3]

4.1.1 Other Related Research

Given the above constraints, there has been little investigation into minimal communication in human-robot interaction; efforts have largely concentrated on achieving common ground through highly targeted anthropomorphism. Given that deliberately anthropomorphic physical design for human-robot social interaction is beyond the scope of the notion of minimal communication being defined here, it is not necessary to summarize the many such projects — see [99] for an overview of robot morphological design and a comprehensive initial set of references on robotic facial expression, as well as an explicit acknowledgement that the minimal criteria for a robot to be social remains an unanswered question. Besides the widespread use of communicative facial expression, a number of these projects demonstrate that natural language use is not a precondition for human-robot communication — for instance, the best known robot example, Kismet, had no formal language skills but did feature exaggeratedly human-like features and underlying emotional models[35]. Body language (see Chapter 6) has been used to communicate emotions through ballet-like gestures and movement[200] and through expressive walking motion[173]. Pre-linguistic instruction of robots has also been demonstrated, such as teaching a robot a rudimentary vocabulary concerning its perceptual mechanisms[23].

Several projects have featured otherwise nonanthropomorphic robots equipped with graphical renditions of human faces, many of which process natural language cues, e.g. [268, 109]. One such project did set out to explicitly examine minimal requirements for social interaction, however their working definition of minimal requirement was minimal "human-like behaviors", which is clearly substantially less minimal than that being discussed here[54]. This study did produce results showing that the robot directing its gaze towards prospective human interactors had no effect on success in attracting people to interact with it, potentially invalidating the use of directed gaze as a minimal communicator of interactive intent. However the authors do cite problems with the tracking behavior as a possible explanation, and the study had other flaws that nay have led people to conclude that the robot was less than minimally interactive regardless of its gaze behavior. In general I would characterize these efforts as more appropriately covering minimal anthropomorphism for HRI; investigating the baseline at which a human might consider a robot to have human qualities, rather than the ultimate baseline of genuine interactivity.

[3] Imagine a robotic insectoid 'Lassie'.

Areas in which minimal communication have received more extensive attention in the context of HRI are those in which the interactivity of the robot is presupposed, and the quantity to be minimized pertains to a particular aspect of the human's cognition. This is typically encountered in the context of collaborative human-robot partnerships or teams. Since such activities can make it difficult for the human to accurately assume the perspectives of the robot participants and aggregate distributed state information, one aspect to be minimized is the amount of uncertainty or error in the human's situational awareness[95, 217]. This is a concern in many applications of human supervisory control, such as teleoperation; the specific HRI considerations have received most attention in the fields of military human-robot teams, swarm robotics and robotic search and rescue (e.g. [265, 2, 206, 309, 55]).

While the communication involved in these situations is by no means minimal in the sense under discussion here, interface efficiency is a major consideration in order to satisfy concerns of operator mental workload[256]. Indeed, in cases where the human-robot relationship is one of supervised autonomy[60], another cognitive aspect that researchers have sought to minimize is the "interaction time" (the expected amount of time that the human must interact with the robot to bring it to peak performance) for a given "neglect time" (the expected amount of time that a robot can be ignored before its performance drops below a threshold)[71]. Some awareness therefore exists in the human-robot interface community of the potential benefits of "natural" interfaces that may incorporate efficiently compressed minimal elements that require low levels of cognitive load on the part of the human. A good example is the use of imprecise physical "nudges" to help influence a robot's inferral of the human's intentions[294].

The most relevant work in this area is recent and in its initial stages. Researchers at the University of South Florida have suggested that due to human expectations even tank-like search and rescue robots could benefit from social intelligence despite their physical nonanthropomorphism and minimal communicative capabilities[98], and have proposed a set of prescriptive mappings to guide the selection of minimal communication elements for appearance-constrained robots based on proxemic zones[22]. It should be noted that many of these research endeavors come several years after the completion of the work presented in this chapter; however it does offer support for the pursuit of minimal communication, the motivation for which shall now be expanded upon.

4.2 The Value of Minimal Communication

Now that minimal communication has been explained in detail, we can set about answering the obvious question: if minimal communication is so basic, why is it worth considering when we can simply implement more complex interaction schemes based on the higher level communication skills possessed by humans from a reasonably early age? Some of the answers to this are already suggested by the human-animal interaction analogies and related work above. However there are several other justifications for codifying and examining minimal communication; even for robots whose primary mission centers on complex human interaction.

Minimal communication is a baseline condition, and one reason to pursue baseline conditions in robotics is that technological limitations may require it. Such limitations impact human-robot communications on the sides of both sensing and acting. Although great strides have been made in recent years towards unencumbered sensing of human activity such as speech understanding[102, 107] and articulated tracking of the body[81], these techniques are still not sufficiently robust to

many unstructured human environments, which may feature substantially less than optimal lighting conditions or levels of auditory noise. Visual motion capture rigs capable of sensing fine details of human posture, for example, remain large, expensive fixed installations[108], and uncalibrated speech recognition remains confined to auditorily constrained environments such as telephone menus and robotics labs[316].

Methods for communication from the robot to the human similarly encounter technological restrictions depending on the application domain. Designers of mobile robots for operation in constrained environments such as disaster areas may not find it feasible to incorporate anthropomorphic expressive features like faces[22]. Despite the attention it has received, developing humanoid faces that reliably express the subtleties human emotions remains difficult. Noisy auditory environments can interfere with a human's perception of synthesized speech just as they can with a robot's. Although robots can be designed to use manual symbolic sign languages such as the Japanese finger alphabet[129, 205], robotic manipulators capable of fully expressing these are rare, expensive, fragile and may in any case be unsuitable for the robot's regular manipulation tasks.

Endowing a robot with minimal communication skills that are known to be robust within the limits of performance of current technology applied to the robot's particular operating environment ensures an adequate baseline level of interactivity. Furthermore, for robots designed with supraminimal communication skills, the minimal set of communicative behaviors can serve as a prioritized "inner loop" or "lizard brain" that can claim limited processing resources during times of high contention, such as an environment with large numbers of humans present. Such a prioritized behavior set could be likened to the amygdala in humans, which is responsible for fear and certain social and emotional judgements and responses[76, 1]; similarly in a robot it would be responsible for self-preservation (locus of attention and spatial presence) and basic social responses even in the absence of human-style emotions. Just as such atavistic brain components in humans allows graceful performance degradation in times of stress, such a module would allow a robot to revert to an acceptable baseline level of interactive performance when more complex communications fail.

Approaching human-robot communication from a minimal perspective supports a developmental, bottom-up approach to robot design. It is true that the merits of this approach vis-à-vis top-down design for high level task performance are still a matter of open debate in the wider intelligent robotics community, particularly in areas such as rapid interaction prototyping (why have a robot behave like a baby when it can behave like a sophisticated but inelastic search engine?). Nevertheless, it has been strongly argued that insights acquired from ontogenetic development have shown the potential for guiding robot design in directions that may ultimately assist in bootstrapping higher reasoning and learning capabilities[176]. Recent results suggesting that the primate amygdala underpins reinforcement learning by representing the positive and negative values of visual stimuli during learning supports the importance of basic brain structures to higher functioning[228]. Similarly, starting human-robot communication capabilities from minimal "first principles" may allow learning-based interaction methods to develop in directions that more rigidly specified top-down communication skills might prove too inflexible to support.

Minimal human-robot communication is also valuable from the human side of the equation. It has been shown that humans will apply a social model when interacting with a sociable robot[40]; it has also been shown that humans apply social models to other pieces of technology that exhibit much less explicit sociable design[245, 218]. Misunderstandings and frustration occur in human-machine interactions when the human expectations and the actual machine performance or

behavior are mismatched[187, 220]. This can also occur when high level expectations are fulfilled but minimal ones are not, such as in the famed "uncanny valley" theory (Masahiro Mori, from [85]). Misunderstandings in human-human interactions due to the same deviation between expected and actual performance are also common, but are often resolved with less confusion and frustration because absent certain pathologies humans do satisfy the minimal communication constraints. A robot with appropriate minimal communication skills gives its human collaborator a fall-back level of ability to achieve common ground with it. Although it clearly is not an absolute prophylactic, this strong baseline for the human's understanding also offers the best chance of robustness to large differences in the robot's morphology, attitudes and emotions compared with those of the human, which otherwise could contribute to immediate and potentially catastrophic misapprehensions if the human is unfamiliar with them.

Finally, minimal communication is important in the case of human-interactive robots whose nonanthropomorphism is not dictated by technical constraints but rather by deliberate design. One such class of robots is non-humanoid entertainment robots, an example of which is presented in this chapter. Other potential examples are robots for which an anthropomorphic appearance might be inappropriate or "creepy", such as perhaps a surgical robot, or robots whose functionality is likely to be unknown or unexpected, such as a service robot which comes into contact with humans too rarely to warrant human-like interactive features but is still expected to maintain baseline interactivity during the occasions on which such contact does occur. Minimal communication offers a base level of interactive performance that has a relatively low implementation cost yet still provides for consistent satisfaction of human social expectations.

4.3 Minimal Communication in a Large-Scale Interaction

The dimensions of possible human-robot relationships have already been distilled into a set of taxonomies[207]. One of these is entitled numeric relationships, referring to the relative multiplicity of humans and robots in a given relationship, and within this taxonomy is the many-to-one relationship: many humans to one robot. Several applications have been fielded in this area, including museum tour guide robots[284], hospital orderly robots[151], and school robots[144]. Many-to-one relationships place additional demands on interactivity, especially when interactions can not be constrained simply to sequential one-to-one interactions.

Entertainment robotics is a prime source of these relationships. Although sometimes described as an emerging discipline, the construction of lifelike machines for human entertainment has centuries of history, from the earliest ancient mechanical automata to modern animatronics in theme parks and motion pictures[306]. In order for robots designed for many-to-one applications to be truly interactive, they must exhibit robust, compelling behavior while coping with multiple demands on their attention. Example deployments such as in-store displays, entertainment facilities and bars, involving interactions with ordinary humans, become an extension of the many-to-one numeric relationship to its most extreme case, a situation which we entitle large-scale HRI. We define large-scale HRI as consisting of very many humans to one robot, including the likelihood of multiple simultaneous interactions, and in which almost no assumptions can be made about the appearance or behavior of an individual person.

Such a scenario is ideal for a robust minimal communication implementation. Human behavior is sufficiently complex that when humans are simultaneously engaged in tasks other than interacting

with the robot, such as shopping, drinking, or socializing with one another, it is easy for aspects of that behavior (e.g. gestures) to be misinterpreted. The robot must be able to unambiguously interpret certain basic aspects of human communication. Similarly, in unsupervised human-robot interactions in which the humans can not be assumed to have any prior knowledge of the robot's behavior, the robot must have access to an unequivocal set of basic output behaviors that naïve humans can recognize.

The example large-scale human-robot interaction that shall be considered here is one in which a lifelike robot performs or "acts" with humans in an interactive fashion that allows the humans to affect the robot's behavior: interactive robot theatre[42]. Interactive robot theatre with human participants has been suggested as a benchmark test domain for the social interactivity of robots, much as the RoboCup soccer tournament has been developed into a test scenario for multi-robot coördination[152]. Although the notion of robots as performers is not entirely new — small mobile robots with navigational skills have been used in live performances[300], and mobile robots with dialogue skills have performed "improvisational" performances with each other[53] — the incorporation of unstructured human interaction remains largely unexplored. Similarly, although animatronic performers are commonplace in certain spheres, their design criteria tend to be the opposite of that under concern here. We are interested in maximal interactivity from minimal communication, whereas the typical animatronic design provides maximal communication (e.g. through speech, gesture, posture and facial expression) yet ultimately minimal interactivity.

The minimal communication requirement for interactive robotic performance is thus specified in terms of the minimal requirement for "acting" itself: the robot's goal is to appear to be an organic life form rather than a robot, behaving as if it is aware of its human counterparts in a natural and organic fashion. What would a robot need to be able to know and do in order to act organically among actual life forms? Returning to the minimal communication requirements for HRI awareness outlined above, I suggest that a robot capable of convincing lifelike acting requires at least the following minimal skills:

- Recognizing where an agent's communication physically originates and is collected. These are not always colocated; for example, dogs' primary sensors are located in the head, yet they can generate communication with the tail as well as the head-based actuators.
- Recognizing the locations of an agent's actuators that can physically affect the robot. Physical action not intended strictly as communication can still communicate intent, such as threat.
- Recognizing whether an agent is aware of the robot or not; more generally, whether or not the robot is within the agent's locus of attention. This can have an important modulating effect on the reactions to other communications.
- Expressing awareness of the above information via an appropriately timed, scaled and directed bodily configuration or behavior (responsive communication).
- Expressing the robot's own communicative capabilities, locus of attention and spatial presence through exploratory contact attempts (demonstrative communication).

A human (or animal) encountering a robotic actor exhibiting appropriate implementations of the above capabilities should be able to consider, at least initially, that the creature is not in fact

a robot at all, provided that the mechanical design and external appearance was such that it did not betray the illusion of life.

4.4 Public Anemone: An Organic Robotic Actor

In order to test and demonstrate a number of aspects of lifelike interactive robotics, we constructed an interactive robot theatre installation incorporating a single nonanthropomorphic, organic-looking robot as its centerpiece: the "Public Anemone" (Figure 4-1). This robot was then exhibited in the context of a large-scale interaction at SIGGRAPH Emerging Technologies. Due to the constraints inherent in the environment, the communicative abilities with which the robot was designed were examples of minimal communication. The robot's sensor suite for communicative input was based on computer vision, and its corresponding output channel was its own bodily configuration and physical behavior. The robot's "terrarium" environment (Figure 4-2), motor control system and physical appearance were designed to appear highly organic in order to preserve the illusion of life, and are described in detail in [42]. This description will concentrate on the design of the communication and the implementation of the underlying sensing.

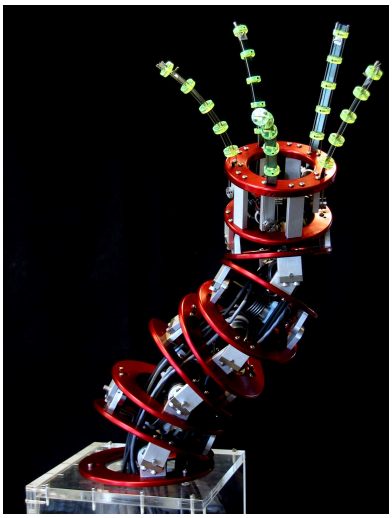


Figure 4-1: The Public Anemone robot, photographed without its skin to show its graceful skeleton.

The constraints of the SIGGRAPH environment were quite restrictive, not including basic practical considerations such as how long the display must be able to run continuously without risking component failure. Due to the constant flow of visitors that could be expected, individual interactions with the robot needed to be short, so the robot had to be immediately convincing as to its interactivity and organicness. The noisy auditory environment and unstructured, dimly lit visual environment restricted the types of sensing that could be performed robustly. Furthermore, very few assumptions could be made about the contents of the visual scene. Unknown numbers of people could be visible at any given time; they may enter and leave the field of view at will; they may possess a range of biometric characteristics such as height and race, and they may be clad in radically differing apparel. Finally, the nonanthropomorphic nature of the robot was chosen deliberately in order to strip away as much as possible of the ingrained social models that humans

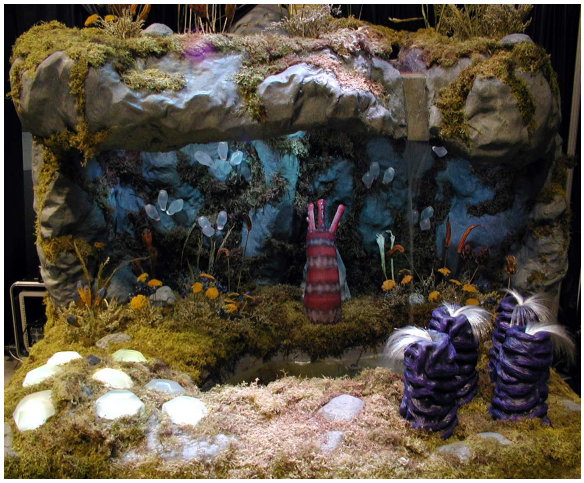


Figure 4-2: The Public Anemone in its interactive terrarium.

would apply when interacting with it, but this also meant that the human participants might be likely to attempt to interact with it in unpredictable ways that the minimal communication suite would have to respond to appropriately.

4.4.1 Design for Minimal Communication

Design of the Public Anemone's sensing and behavioral output satisfied the minimal acting skills outlined in Section 4.3. In humans, the primary communication receptor is the head, and the primary communication generating actuators are the head and hands. This was also convenient for the design of the vision system, as these features are relatively invariant in form between individuals and are unlikely to be obscured by clothing. Furthermore, the presence of a visible face from the robot's perspective indicates that the human is facing the robot and thus the robot is potentially within the human's locus of attention. The robot's sensory system therefore concentrated on the ability to robustly detect and track human faces and hands, satisfying the first three skills. In addition to the constraints enforced by the scene, the sensory processing has a strict real-time constraint, as excessive delays in the robot's reactions can immediately tarnish the interaction as unconvincing. We specified a minimum frame rate of 10 Hz for the vision system. Implementation details of the vision system are given in section 4.4.2.

The Public Anemone's expressive behaviors were designed to communicate the nature and responsiveness of this unusual beast to the humans interacting with it. Most importantly, the Anemone attempted to acknowledge communication from humans, and to establish and maintain contact with them. When the robot detected human faces oriented towards it, indicating that it might be within the human's locus of attention, it responded by communicating to the human that he or she was now within its own locus of attention. It did this by orientating its body and tentacles towards the location of the human face. Furthermore, it communicated its curious and friendly attitude towards humans by adopting a "vulnerable" stance, with its smaller tentacles splayed outwards. This stance was also adopted if the robot detected deliberate attempts by humans to engage its attention using their other primary actuators, the hands, through waving.

When not engaged in responsive communication of this type, the Anemone had a number of behaviors selected according to internal drives. These included "drinking" from its pond, "bathing" in the waterfall or "watering" its nearby plants. In order to demonstrate that human interaction was a high priority for the robot, it continued to track human hands even when a lack of detected faces indicated it was not the immediate subject of anyone's attention. One of its internal drives regulated an urge to initiate communication by applying its interactive stance towards humans who were not engaging it, based on their hand positions. The importance to the Anemone of human contact was reinforced by its willingness to interrupt its other activities in order to interact.

Despite its willingness to interact, the robot also communicated its awareness of human activity and its own personal spatial presence by exhibiting a threat response when people stretched their hands too close towards it, clearly indicating that it did not wish to be touched. This was accomplished with a rapid recoiling motion and the bringing in of the smaller tentacles into a closed "defensive" posture, while shaking its body in a manner intended to evoke the appearance of fear or aggression. Due to genuine concerns about participants handling the robot, after initial testing this postural response was accompanied by an auditory cue similar to a hissing rattlesnake, producing a more effective combined image.

The Anemone's sensing and display behavior suite thus covered all of the required minimal capabilities in order to attempt to act as an organic creature capable of convincing interactions with humans.

4.4.2 Visual Algorithms and Implementation

Due to the fluidity of the visual scene, the Anemone's visual sensing was accomplished by a cascaded arrangement of model-free feature detectors, arranged in a mutually reinforcing fashion similar to that exhibited in [74] for face tracking. A feature detector that has a very high "hit" probability but also a reasonably high "false alarm" probability is essentially useless by itself; but when more than one such detector's results are evaluated in series, the high hit probabilities multiply to a probability that is still relatively high, whereas the lower false alarm products approach zero much more rapidly. Several such detectors are thus able to robustly detect objects sharing conjunctions of their properties. The detectors that were incorporated into this system were stereo-based foreground extraction, human flesh color detection, and face detection. These algorithms are well known computer vision techniques and are presented here for completeness. Extracted hands and faces were parameterized for computational convenience, and these descriptive representations tracked over time. A conceptual diagram of the overall system is shown in Figure 4-3.

In order to facilitate ease of deployment and integration into an overall framework for robot behavior, the vision system was designed to be modular and support multiple concurrent instances. Each instance processes video data from its own stereo camera pair independently and sends three-dimensional hand and face data to a coördinating server over the network. The server is then able to either fuse these data or make use of them on their own. This allows essentially unrestricted flexibility in camera placement.

The system was also designed for rapid calibration at the place of operation. The camera systems themselves must be precalibrated once either before or during deployment, but the only geometric calibration that is required to occur during the on-site assembly of the installation is the recording of the camera positions according to a system of absolute world coördinates. Calibration

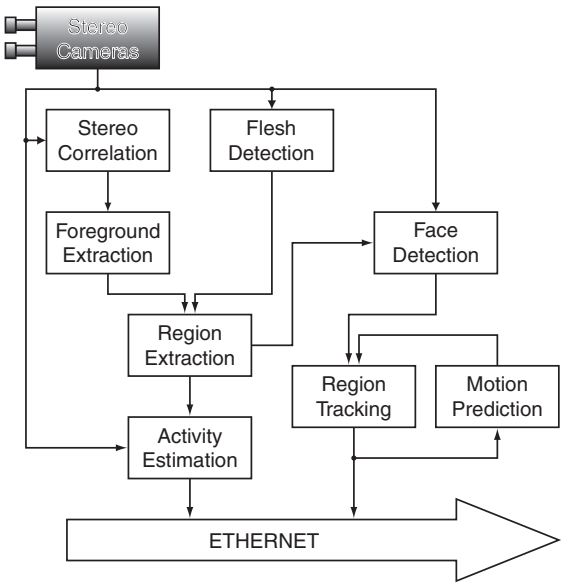


Figure 4-3: Data flow diagram of the vision system.

of the feature detectors is primarily a matter of optimizing for the specific lighting conditions at the site, which is a brief and largely automated procedure.

The Public Anemone vision system that was ultimately used at the SIGGRAPH Emerging Technologies exhibition was deployed in two instantiations. Each ran on a standalone IBM IntelliStation M-Pro dual processor Pentium III workstation. The layout of the vision system with respect to the environment can be seen in Figure 4-5. Both stereo camera systems were prefabricated, fixed-baseline rigs from Videre Design: a Mega-D faced directly forward over the robot, while a Dual-DCAM was suspended above the installation pointing directly downwards. The cameras were concealed in the environment so that the humans interacting with the robot were not aware of them, preserving the illusion of natural interaction with a self-contained organic entity.

4.4.2.1 Stereo Processing

The foreground of the scene, containing all visible humans, was initially extracted via background subtraction from the range data produced by the stereo cameras.[4] This technique is significantly more robust to illumination and video noise than background subtraction directly within the raw image intensity domain[96, 74, 137]. Background subtraction was performed with a standard unimodal Gaussian noise model: the mean values and standard deviations of each pixel in the depth map was learned over a few hundred iterations, and from that point on depth pixels were thresholded based on their deviation from the mean value for that location. A threshold value of three standard deviations was used, which is relatively typical for this application. It was not necessary to consider more sophisticated statistical noise models, such as bimodal distributions[123], as these

[4] Although the Videre Design stereo cameras are supplied with a stereo correlation library, the Public Anemone vision system used a custom library developed by Dr David Demirdjian of the MIT CSAIL Visual Interface Group for performing 2-pass greyscale sum of absolute differences (SAD) correlation followed by disparity propagation, in real time. Dr Demirdjian was also a principal co-architect of the rest of the vision system described herein.

are mainly used to account for image variation of a periodic nature. This is not a difficulty with which we generally are forced to contend in the depth domain.

The foreground map resulting from this technique usually suffers from small holes due to textureless patches on the images that cause uncertainty in the stereo correlation. This problem was alleviated with a morphological close operation, which proved sufficient in light of the fact that the stereo foreground map is not used alone but instead acts as a mask for later processing (a more sophisticated method of generating a dense stereo map for foreground extraction can be found in [73]). It is also worth noting two further caveats regarding the quality of the foreground map. First, the minimum effective distance of the stereo system depends on the maximum pixel disparity able to be computed, which is limited by the number of correlations that can be performed in real time. In the Public Anemone deployment 32 pixels of disparity were used, corresponding to a minimum effective distance of around 4 feet. Second, the background subtraction phase is the only part of the system that requires a static camera and thus would not be suitable for mounting on a mobile robotic platform. However, in such circumstances this computation is able to be replaced by a depth threshold, which can offer comparable performance over typical interaction distances.

4.4.2.2 Flesh Detection

The flesh classifier used in this system identifies regions of human skin based on its characteristic chrominance, using color information from the intensity images that has been normalized to remove the luminance component. This technique has been demonstrated in a variety of person-watching applications, such as intelligent spaces and face tracking (e.g., [12, 310]) and has been shown to be resistant to variation in illumination and race when coupled with a high-quality sensor[48].

$$\begin{aligned} R_{NORM}(r,g,b) &= r/(r+g+b) \\ G_{NORM}(r,g,b) &= g/(r+g+b) \end{aligned} \tag{4.1}$$

Initial normalization of the pixel color channels, as in Equation 4.1, allows the disposal of one channel as redundant information. A model for the appearance of human flesh is then trained in the space described by the remaining two channels. A simple histogram approach has been shown to be successful, but the final Public Anemone system used a statistical model similar to that presented in [119]. The training system was presented with a video sequence of identified human flesh under the desired conditions, using this to model the color as a unimodal, bivariate distribution by collecting the means and covariances of the two channels in the sequence of training pixels. This distribution was then used to convert a given pair of input values into a probability that the pixel corresponds to a region of human flesh, using the thresholded value of the Mahalanobis distance of the pixel to the distribution. Best results were achieved by making this probability conversion have a direct exponential fall-off with the Mahalanobis distance, M. The base probability B enforced by the Mahalanobis threshold M_T was set to be 10%, and the threshold itself worked well at 3.5. The conversion function is summarized in Equation 4.2.

$$P_{FLESH}(x,y) = \begin{cases} B^{\frac{M}{M_T}} & | \quad M \leq M_T \\ 0 & | \quad M > M_T \end{cases} \tag{4.2}$$

The resulting thresholded probability mask was cleaned of spurious small positives with a morphological open operation before being compared against the foreground mask generated by the stereo process. The result was a reliable probability map for hands and faces.

At this point it was computationally convenient to isolate areas of high hand and face probability into parameterized regions. Each of the most promising regions was sequentially extracted by selecting the point of maximum likelihood on the probability map, and performing a flood fill to determine its extent. This cluster of pixels is then subtracted from the map, and an optimally bounding ellipse calculated. This representation provides five intrinsic values for later matching: 2-D location, major and minor axis lengths, and rotation angle. In addition, an average depth is computed for each region, from the disparity map.

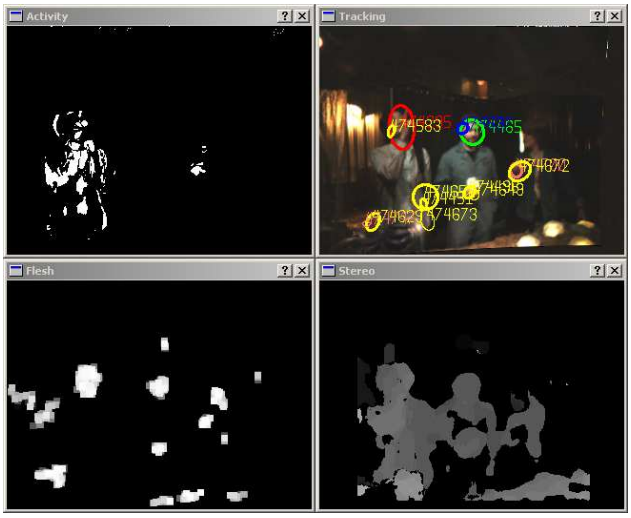


Figure 4-4: Display output of the Public Anemone vision system in operation, showing motion, flesh and stereo maps, and face and hand detections of varying trust levels.

4.4.2.3 Face Detection

To discriminate between hand regions and face regions, the fast face detector developed by Viola & Jones[293] was used.[5] Although this algorithm could be considered to be operating on a high-level, compound feature, in reality it is a model-free approach. Unlike face detectors that consider the relationships between facial subfeatures (e.g. [311]), the Viola-Jones detector uses a cascade of simple rectangular classifier filters that operate in series on an image derived from the greyscale pixel intensities. These classifiers are trained from a large data set and selected on the basis of high hit rates. When hundreds of them are chained sequentially, the result is a hit rate that remains high and a false positive rate of below 1%.

Although this algorithm was among the fastest of contemporary face detectors, it looks for faces at a number of size scales and was thus still computationally prohibitive to run over the entire image at the same frame rate as the rest of the system. The detector was therefore only run within a

[5]Since this work predates the public inclusion of this face detector in OpenCV, I am grateful to Mike Jones for generously allowing me to use his early face detector source code.

proportional region of interest surrounding each candidate ellipse. When the face detector detected the presence of a face within an ellipse, the object was labeled as a face for future processing.

4.4.2.4 Tracking

Each frame thus produced a list of candidate detections to be tracked over time. Each candidate had an associated feature vector that consists of its ellipse parameters, its depth, and whether or not it is a face. The immediate first- and second-order position differences were also stored for each region, to estimate velocity and acceleration, and each region was assigned a unique numeric identifier.

The motion of human body parts is problematic for many tracking techniques (for a comprehensive introduction to classic approaches to the tracking problem, see [18]), because it tends to be highly non-linear and not well described in terms of a tractable number of motion models. Popular techniques based on linear or linearizable models, such as the regular and extended Kalman filters[143, 138], thus have their assumptions invalidated (but are nevertheless sometimes used in this application by resorting to imprecise system models). A further problem with the use of these techniques is the low resolution of the camera image, which largely invalidates Gaussian noise model assumptions also. Multiple-model techniques, such as Interacting Multiple Model filters[190], can be used instead of attempting to describe the non-linearity of human motion with a single model, but the problem of dividing this motion up into appropriate sub-models remains. Furthermore, this application has the added complications of multiple targets, clutter caused by possible false detections, and frequent false negatives.

For the sake of reduced complexity and computation, a custom forward-chaining tracking algorithm was employed to track the hand and face candidate regions, that incorporated a single non-linear predictive motion model and a decision tree based on a set of object matching criteria. The algorithm is conceptually simple but allowed us to track up to thirty maximum candidate regions on average PC hardware (for the time) without incurring the overhead of more processor-intensive popular schemes such as sample-based filtering (for example, Condensation[134]).

Each region present in the list of regions at time $t-1$ had its feature vector perturbed according to our motion model. The principle quantity considered in the non-linear motion model was the negative of the second-order position difference. The rationale was that if an object is seen to accelerate, principles of organic movement suggest that it is most likely to decelerate rather than continue to accelerate. The modified feature vector was then compared against those of the new detections to determine if any matches are possible within a nearness threshold. If so, all new matches are ranked according to a weighted distance estimate over all parameters. The closest match is selected, the old object's feature vector is updated with the quantities contained in the new one, and it is promoted to the current object list with its identifier intact. This procedure is summarized in Equation 4.3 for object parameter vector $\vec{p} = [x, y, z, w, h, \theta]$, search range threshold vector $\vec{\tau}$ and parameter weight vector $\vec{\omega}$ (for example, position is weighted more heavily in the comparison than rotation angle). New detections that are not matched are also promoted, but are not considered reliable until matched in future frames.

$$
\begin{aligned}
&\forall\ \alpha \in \text{ObjectList}_{t-1}: \\
&\quad \vec{p}_{\alpha_{t+1}} = \vec{p}_{\alpha_t} + \vec{p}_{\alpha_{t-1}} - \vec{p}_{\alpha_{t-2}} \\
&\quad \text{if } \exists\ \beta \in \text{ObjectList}_t \ s.t.\ |\vec{p}_\alpha - \vec{p}_\beta| < \vec{\tau}: \\
&\qquad \forall\ \beta \in \text{ObjectList}_t: \\
&\qquad\quad \alpha \leftarrow \arg\min \|(|\vec{p}_\alpha - \vec{p}_\beta|) \bullet \vec{\omega}\|
\end{aligned}
\tag{4.3}
$$

Despite the high true positive rate, tracking losses do occur. For this reason, old objects that are not matched were still retained in the search list while a decay period expires. The proportion of time spent decaying as compared to that of being actively tracked was also incorporated into the reliability statistic for the object.

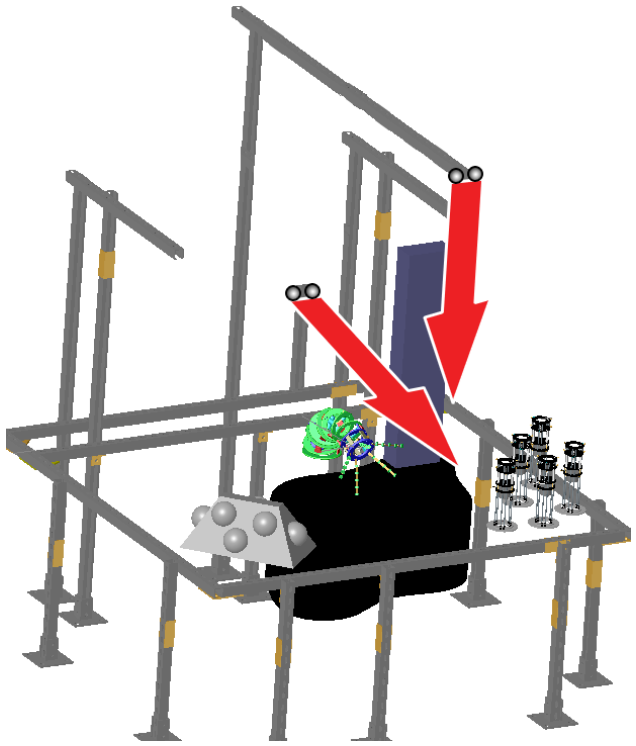


Figure 4-5: Skeletonized view of the Public Anemone terrarium, showing the placement and orientations of the rear and overhead camera setups.

One particular type of human movement that can present problems for tracking is fast, periodic motion — a feature found in the common, minimal human communicative behavior of waving. When this occurs, the hand is likely to appear as a large, reasonably static, blurry region rather than a sequence of discrete positions. For this reason, an activity score was also computed for each region based on the image intensity difference from the previous frame. The activity score was sent, along with the object identifier, estimated 3-D position and reliability statistic, and a frame timestamp, to the robot behavior server. Each object's details were encapsulated in a single UDP packet; it was assumed that all packets would make it to the destination within one frame time, an assumption that proved valid in the Public Anemone setup.

4.4.3 Results and Evaluation

The Public Anemone interactive robotic terrarium was operated for 8 hours per day over 5 contiguous days at the SIGGRAPH 2002 Emerging Technologies exhibit. Visitors numbered in the hundreds, frequently interacting with the installation in large groups, rendering the interaction truly large-scale.

Under these conditions it was not possible to collect interaction data via any automatic mechanism. However a voluntary self-report questionnaire was distributed[6] to participants following their interaction with the terrarium. Each question asked the respondent to rank their experience on a 7-point scale from 1 (weak) to 7 (strong). A total of 126 surveys were completed by 79 men, 45 women and 2 who did not state their gender. The respondents ranged in age from 16 to 67 years. A diverse occupation background was represented including people from academia, the arts (e.g. music, animation), engineering (e.g. video game designers), and the media (e.g. television and film). The most relevant scores pertaining to the interactivity and realism of the Public Anemone as supported by its minimal communications capability are reported in Table 4.1, showing the generally positive results.

Quality	**Mean**	**Std. Dev.**
Engaging	5.5	1.2
Liveliness	5.4	1.3
Responsiveness of Robot	5.2	1.3
Naturalness of Interaction	5.2	1.5

Table 4.1: Selected relevant results from Public Anemone user experience questionnaire. Participants evaluated each quality on a 7-point scale (1 weak, 7 strong).

The specific techniques used in this work, such as the visual processing algorithms, are now relatively common in HRI, and their implementation *in vacuo* does not represent a contribution to the field. However their synthesis, within the context of both the novel minimal awareness definition suggested herein and the constraints of the non-anthropomorphic robot theatre setting, does indeed demonstrate a new view of communicative interaction design. Baseline communication and awareness skills for sociable robots have not been stipulated in the literature let alone sufficiently explored, but the success of the Public Anemone installation shows that careful design and assembly of minimal sensory and expressive components can support an interactive experience that is convincingly natural, responsive and engaging (as judged by naïve participants). The project, including its robust deployment for several days in a large-scale venue, thus represents a concrete and unique contribution.

It also demonstrates the need for further discussion, research and standardized descriptions of factors in human-robot communication that are not expressed in terms of high level task performance or anthropomorphic analogies. The definition of minimal awareness suggested here has not been formally accepted by the HRI or CSCW communities — but nor have many other definitions of awareness published in the field. It therefore itself represents a minor but genuine addition to this discussion.

[6]The questionnaire was written and administered by Cory Kidd.

Chapter 5

Multimodal Spatial Communication

Humans and robots interacting with one another in a shared workspace need to be able to communicate spatial information. In particular, if the workspace contains objects besides the human and the robot (and for most non-trivial examples of collaboration, it will) they need to be able to communicate about those objects spatially. That is, the human and the robot need to be able to refer to objects not just in terms of what they are, but where they are — and the latter can often be also used to communicate the former. A common spatial frame of reference — the human and the robot being "on the same page" as to which object is where — is crucial to avoiding misunderstandings and achieving accurate and efficient coöperation.

One of the important mechanisms underlying the successful achievement of a common spatial frame of reference is joint visual attention[259, 202]. The creation and maintenance of joint visual attention not only ensures that the human and the robot agree on what is important, but that they also agree on its spatial position relative to themselves. Given adequate communicative and perceptual abilities, it is possible to achieve joint visual attention via a variety of unimodal mechanisms. Language, whether written or spoken, can be used to precisely describe the location of an object to which one is referring. Different types of gestures, including sign language and tactile gestures, can be used to demonstrate the spatial locus of attention. However it is often more natural and intuitive for humans to indicate objects multimodally: through a combination of speech and deictic gesture.

In order to support natural human-machine interactions in which spatial common ground is easily achieved, it is therefore desirable to ensure that deictic reference be a robustly implemented component of the overall communication framework. Indeed, having been identified early on as a primary candidate metaphor to be transferred to human-computer interaction, deixis — in particular the "concrete" or "specific deictic", using pointing to refer to a specific object or function — is now a staple of interfaces involving 2-D spatial metaphors. Unsurprisingly, it has also been an attractive communications mode in human-robot interaction. However, due to its overall conceptual simplicity, in many cases the central component is implemented in an ad hoc fashion: namely, the process of correctly determining the object referent from a combination of pointing gestures and speech.

This is likely a result of considering the task in deceptively simple geometric terms: using the human arm to describe a vector in space, find the closest object it encounters. However,

this approach has a number of limitations. Excessive focus on the gestural component fails to correlate it to speech with sufficient tightness, so that the necessary user behavior is not natural. Moreover, undue faith in human pointing accuracy can necessitate the imposition of a highly deliberate pointing style in order to ensure accurate results. The problem itself contains inherent ambiguities, such as the underspecification of the gesture's ultimate destination point, nebulousness in the decoding of references to compound objects, and the frequent presence of encapsulating context for the reference that affects how it should be resolved. As a result, deictic reference systems in three dimensions are often linked to simplified interaction scenarios.

I have therefore revisited the theory and execution of deictic object reference in order to develop a system for achieving spatial common ground in a scalable and robust fashion during collaborative human-robot activities[51]. In this chapter I will first summarize the use of deictic reference and its application to human-robot interaction. I will then present a novel, modular spatial reasoning framework based around decomposition and resynthesis of speech and gesture into a refined language of pointing and object labeling. This framework supports multimodal and unimodal access in both real-world and mixed-reality workspaces, accounts for the need to discriminate and sequence identical and proximate objects, assists in overcoming inherent precision limitations in deictic gesture, and assists in the extraction of those gestures. Finally, I will discuss an implementation of the framework that has been deployed on two humanoid robot platforms.

5.1 Deictic Spatial Reference

Concrete deictic gestures are powerfully expressive and very common. Human-human interaction studies have showed that over 50% of observed spontaneous gestures to be concrete deictics[115], and that concrete deictic gesture can effectively substitute for location descriptions in establishing joint attention[175]. Unaugmented human-human use of deictic gesture has two important characteristics which carry over to human-robot interaction. First, it is only successfully used in cases of what Clark and Marshall refer to as "physical copresence"[65] — both participants must be able to view the referent in the situation in which the gesture occurs.

The robot's pre-existing knowledge of the spatial state of the world thus should contextualize the gesture; in fact, the nature of human pointing behavior makes this a necessity. The precise point indicated by the human (the *demonstratum*) is in most cases spatially distinct from the object the human intends to indicate (the *referent*)[66] due to various factors including simple geometric error on the part of the human (e.g. parallax), the human's desire not to allow the gesture itself to occlude the robot's view, and the fact that pointing gestures in 3-D contain no inherent distance argument. Instead, the demonstratum can typically only constrain the set of potential referents. This argues for a "spatial database" approach in which deictic gestures are treated as parameterized spatial queries.

Second, like most gesture, deictic gesture is closely correlated with speech (90% of gesture occurs in conjunction with speech[193]). Natural language itself contains deictic expressions that can be disambiguated with the help of gesture, and similarly deictic gestures are usually resolved in the presence of their accompanying spoken context. Pointing gestures unaccompanied by contextualizing utterances are rare, and depend on other narrow constraints to allow them to be understood, such as specific hand configurations (index finger outstretched) along with situational context (e.g. to distinguish a pointing up gesture from a symbolic reference to the numeral 1). In

the typical case the robot should be able to use the accompanying speech both to assist in the spatio-temporal isolation of the gesture itself, and to constrain the demonstratum to the referent in cluttered or hierarchical situations.

5.1.1 Related Work

The "Put-That-There" system of Bolt in 1980 essentially set the standard for copresent deictic gestural object reference in human-machine interaction[31]. Conceptually, the task of this system is similar to that of the human-robot collaboration task: to refer to objects by pointing and speaking. This system used a Polhemus tracker to monitor the human's arms, and the spatial data to be managed consisted of simple geometric shapes in a 2-D virtual space. Most of the subsequent advances within this task domain concern improvements to the underlying components such as tracking and speech recognition.

For a comprehensive summary of work on the visual interpretation of hand gestures, including deictic gestures, see [229]. To overcome technological limitations in natural gesture analysis, investigations into multimodal interfaces involving deixis were often restricted to constrained domains such as 2-D spaces and pen-based input (e.g. [198]). The deictic components of these efforts can be summarized as enabling this type of gestural reference in some form as part of the interface, rather than tackling problems in determining the object referent from the gesture.

Several research efforts chose to concentrate on the object referent primarily in order to use it as contextual information to assist in recognition processes. Kuniyoshi and Inoue used the object context to aid action recognition in a blocks world[162]. Moore et al. related object presence to the trajectory of the hand in order to provide action-based object recognition[203]. Strobel et al. used the spatial context of the environment (e.g. what object lies along the axis of the hand) to disambiguate the type of hand gesture being performed in order to command a domestic service robot[278]. Similarly, the first deictic reference system I developed to support human tutelage of our humanoid robot Leonardo used the presence of a visually identified button object to confirm a static pointing gesture after being suggested by the hand tracking system[43]. Nagai reports using object information as a feedback signal in a system for teaching a robot to comprehend the relation between deictic gesture and attentional shift[209]. In contrast, the work presented here is primarily concerned with robustly connecting the demonstratum with the desired referent.

The majority of work in determining the object referent from deixis has been directed towards using gestural information to disambiguate natural language use. Kobsa et al. used gestural information in the form of a mouse pointer in a 2-D space to resolve deictic phrases[154], and more recently Huls et al. used similar mouse-based pointing as part of a system to automatically resolve deictic and anaphoric expressions[131]. Koons et al. resolved deictic expressions using visual attention in the form of both pointing and gaze direction[156]. Similar efforts are beginning to appear in the human-robot interaction literature; for example, Hanafiah et al. use onboard gaze and hand tracking to assist a robot in disambiguating inexplicit utterances, though only results for the gaze component are reported[121]. In contrast to this work, these systems concentrate on disambiguating speech as it occurs, rather than augmenting the shared spatial state with data for future object reference, such as object names.

Research concentrating primarily on determining and managing object referents from pointing is less common. The ability to use deictic gesture over a range of distances is an attractive

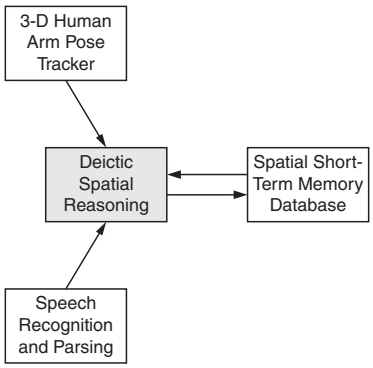


Figure 5-1: The functional units of the object reference system and their data interconnections.

feature in virtual environments (VEs), and Latoschik and Wachsmuth report a system that classifies pointing gestures into direction vectors that can be used to select objects, but do not tackle object discrimination[166]. A later VE system reported by Pfeiffer and Latoschik is more closely aligned with the efforts described in this thesis, but as above focuses more on the disambiguation of speech than gesture, and resolves multiple object reference with relative speech references rather than further gesture[236]. Hofemann et al. recently report a system dedicated to simultaneously disambiguating pointing gestures and determining the object referent by combining hand trajectory information with the presence of objects in a "context area", but does not deal with discrimination within homogeneous or heterogeneous collections of multiple objects. In contrast, my system is specifically designed for working with multiple, potentially visually identical objects arranged within the margin of error of human pointing and visual tracking.

5.2 Deictic Grammars for Object Reference

The framework for deictic object reference developed in this thesis is a distributed, modular approach based on largely independent functional units that communicate by message passing. This enables most modules to be simultaneously available for other tasks in addition to object reference. Individual modules may be executed on different computers for performance reasons. The underlying metaphor is one of using a "deictic grammar" to assemble some combination of gestures and object names and relationships into queries to the robot's spatial database.

5.2.1 Functional Units

The four functional units of the framework are a vision-based human tracking system for pointing gesture extraction, a grammar-based speech understanding system, a spatial database, and the main deictic spatial reasoning system. The units and their data interconnections are shown in Figure 5-1. The first three components are treated in a generic fashion; they can and have been represented by different implementations that just need to satisfy the essential data requirements.

The human tracking system is a 3-D model-based tracker capable of real-time extraction and reporting of the human's arm or arms. While desirable, it is not strictly necessary for the tracker to also detect pointing "gestures" by incorporating the configuration of the hand into the model.

Distinguishing pointing from non-pointing is bootstrapped with the aid of the speech module. I have preferred to use an untethered vision-based tracker to maintain as much as possible the naturalness of the interaction, but this is also not a requirement. The local coördinate system of the tracker is converted into the robot's egocentric coördinate system.

The speech understanding system incorporates speech recognition and parsing according to a predefined grammar that is able to be tagged at points relating to deixis, such as direct references to objects (names and deictic expressions such as "this" and "that") and instructions indicating an object context (such as naming something). The system must thus provide access to the tags activated for a particular parse.

The spatial database primarily stores the names, locations and orientations of objects in space. It is independently updated by the robot's own perceptual and mnemonic systems. Additional levels of sophistication such as more descriptive metadata (object types and hierarchies of compound objects) and the ability to query the database by spatial partition are helpful but not required. For simplicity of implementation during development I have encoded the basic type of an object as a prefix of its name.

The core of the system is the deictic spatial reasoning module. This unit continuously monitors the output of the human tracker, and uses the tag signals from the speech parsing to extract pointing gestures and assemble queries for potential object referents from the spatial database. Candidates returned from the spatial database are matched and confirmed by this unit, and updated result information, along with the pointing gestures themselves, are posted back to the spatial database for subsequent access by the robot's attentional mechanisms.

5.2.2 Gesture Extraction

Hand gestures consistently adhere to a temporal structure, or "gesture phrase"[147], comprising three phases: preparation, nucleus (peak or stroke[194]), and retraction. Visual recognition of deictic gestures is limited to the nucleus phase, by identifying the characteristic hand configuration (one finger outstretched) or making pose inference (arm outstretched and immobile). However these characteristics are frequently attenuated or absent in natural pointing actions. Conversely, as discussed earlier deictic gestures are almost always accompanied by associated utterances. I therefore chose to isolate pointing gestures temporally rather than visually.

Speech is synchronized closely with the gesture phrase, but the spoken deictic element does not directly overlap the deictic gesture's nucleus in up to 75% of cases[224]. However, considering the entirety of both spoken and gesture phrases, the overlap is substantial and predictable. Marslen-Wilson et al. observed that pointing gestures occurred simultaneously with the demonstrative in the noun phrase of the utterance, or with the head of the noun phrase if no demonstrative was included, and that no deictic gestures occurred after completion of the noun phrase[186]. Kaur et al. similarly showed that during visual attention shifts, the speaker's gaze direction began to shift towards the object referent before the commencement of speech[146]. These results were matched by our own informal observations, in which subjects frequently commenced pointing prior to both the deictic and noun components of their utterances, during human subject experiments using our previous pointing system[45].

To extract pointing gestures temporally, I therefore consider "dynamic pointing", incorporating the preparation phase as well as the nucleus, in addition to the more traditional "static pointing".

This also makes it possible to successfully recover deictic gestures in which the nucleus is not static, i.e. motioning towards an object. Because it can be assumed with reasonable confidence that the pointing gesture occurred during or shortly before the user's speech, the spatial reasoning system keeps a history buffer of the arm tracking data. When a deictic phrase component is received from the speech understanding system, this buffer is analyzed to extract the best estimate of the gesture type and location. For implementation details, please see Section 5.3.

5.2.3 Object Reference

First among the ten myths of multimodal interaction identified by Oviatt is that users of a multimodal system will always tend to interact multimodally[223]. In fact, users frequently prefer unimodal interaction, particularly in the case of object reference — there should be no need for the human to point to an object each time the robot's attention is to be drawn to it. I therefore treat the traditional "point-to-select" usage as a special case of a more general "point-to-label" metaphor. Once an object has been labeled for the robot, the human can refer to it unimodally by name, as well as making partial reference by name to constrain potential referents of future deictic gestures.

Object labels are also used to allow ordering of multiple objects for tasks in which the robot must attend to them in a particular sequence. Many collaborative tasks that might be performed with robotic assistance, such as assembly of a complex object from components, require attention to otherwise identical objects in a specific order, yet this has not been widely explored in the design of tools for human-robot interaction. By incorporating a sequence number into each object label, the robot can generate appropriate behavior from future unimodal references (e.g. "first", "next", "last").

As already discussed, object reference from deictic gesture must content with a number of ambiguities. For example, the *pars-pro-toto deictic*, in which a superordinate object is indicated by pointing to an individual subcomponent, and the opposite case, the *totum-pro-parte deictic*, in which the superordinate object is used to refer attention to a specific subcomponent. Moreover, 3-D pointing gestures have no inherent distance constraint, so particular spatial layouts can present perspective ambiguities that are difficult to resolve geometrically.

I have tackled these problems by allowing nested object references in which the user can first deictically indicate a parent object to constrain further references until the constraint is removed. When such a hierarchical object reference is set up, the system restricts potential matches to objects physically or conceptually dependent on the constraining object. This can occur both at the spatial database level, in terms of hierarchically constructed compound objects, and at the spatial reasoning level, in which a plane defined by the horizontal axes of the parent object is used to constrain the distance implied by the pointing gesture vector.

As mentioned above, pointing gestures themselves are also treated as virtual object references and thus can be posted to the spatial database as objects that are referred to by themselves. This provides a seamless method of providing access to pointing gestures to other attentional mechanisms (which may not be interested in object reference directly) and supports potential future extensions to reasoning about the spatial behavior of the human.

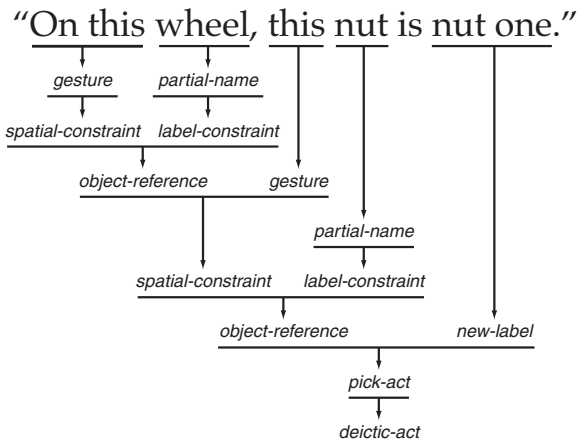


Figure 5-2: Parsing a sentence containing a nested object constraint and multiple deictic gestures into a single pick action intended to focus the robot's attention on an object and label it for potential unimodal reference and sequencing.

5.2.4 Deictic Grammar

Deictic expressions are a grammatical component of natural language, but non-verbal communication or "body language" is not strictly a language — it does not have discrete rules and grammars, but does convey coded messages that humans can interpret[177]. The framework developed in this thesis is based on synthesis of the spoken and gestural elements of deictic reference into a "deictic grammar", a formal language for physically and verbally referring to objects in space. Complex speech and movement are decomposed into simpler features that are sent to the deictic spatial reasoning system to be parsed according to the following ruleset, where the vertical bar '|' represents either-or selection and square brackets '[]' represent optionality:

deictic-act $\longrightarrow$ **pick-act** [**go-act**] | **place-act** | **delete-act**

pick-act $\longrightarrow$ **object-reference** [**new-label**] [**pick-act**]

place-act $\longrightarrow$ **pick-act object-reference**

delete-act $\longrightarrow$ **object-reference**

object-reference $\longrightarrow$ [**spatial-constraint**] [**label-constraint**]

spatial-constraint $\longrightarrow$ [**gesture**] [**object-reference**]

label-constraint $\longrightarrow$ [**full-name** | **partial-name**]

In this deictic grammar, **pick-act** indicates an attention directing command with associated referent relabeling, **place-act** indicates a command to move one or more objects, **delete-act** indicates a command to remove an object from the spatial database. The **pick-act** term has been designed recursively in order to support multiple simultaneous referent resolution. In cases of multiple resolution, gestures are held in a queue and only matched against results from the spatial database when the system is best equipped to do so. A special end-of-sequence marker, **go-act**, is therefore necessary to instruct the system to drain the queue of deictic references in these cases.

The **spatial-constraint** term is also recursive via **object-reference**, in order to support nested object reference as described in Section 5.2.3. See Figure 5-2 for an example parse of a typical

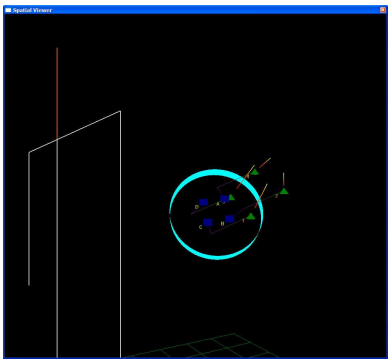


Figure 5-3: Example screenshot from the 3-D visualizer showing four gestures being simultaneously matched to four objects constrained by a parent object, including the actual pointing error in each case.

compound deictic act, showing how a combination of multiple gestures and object references can be used to relabel a single specific object. The deictic grammar imposes some restrictions on the spoken grammar that can be used to drive it. This is acceptable for the HRI applications for which the system has thus far been developed because the speech recognition engine itself also requires a pre-written grammar, so it simply requires that this framework to be taken into account when designing the space of utterances the robot can understand.

The top level commands were chosen in order to primarily support object reference in the real world, with extensions to support mixed-reality workspaces (i.e. where objects can be moved by the spatial database directly, and where object deletion makes sense). I chose not to specifically incorporate object creation, as this system is designed for the spatial reference of objects that the robot knows to be physically present. None of the current module interconnections specifically deals with objects the robot knows about but which are not present, so the spatial reasoning system can not post arbitrary objects to the spatial database. This functionality could be added by having another module read posted gestures from the spatial database and communicate with the speech system directly.

5.3 Implementation of the Deictic Reference Framework

Development of an implementation of this framework for deictic spatial reference took place in the context of two humanoid robot projects. The first was the NASA Robonaut program, in which I participated as part of a multi-institutional collaboration. The second was our own humanoid HRI testbed robot, Leonardo. The spatial reasoning system was written in C++ and runs on an IBM workstation running Windows 2000. It incorporates a 3-D visualizer of objects and pointing gestures, written using OpenGL (Figure 5-3).

The functionality of the other modules has been provided by a number of other systems. During my involvement with the Robonaut project, visual tracking of the human was performed by the system developed by Huber[130]. Speech recognition was accomplished using ViaVoice and the natural language understanding system Nautilus developed at the Naval Research laboratories[232]. The spatial database used was the Sensory Egosphere (SES) developed at Vanderbilt University[234]. Intermodule communication used Network Data Delivery System by Real-Time Innovations, Inc.

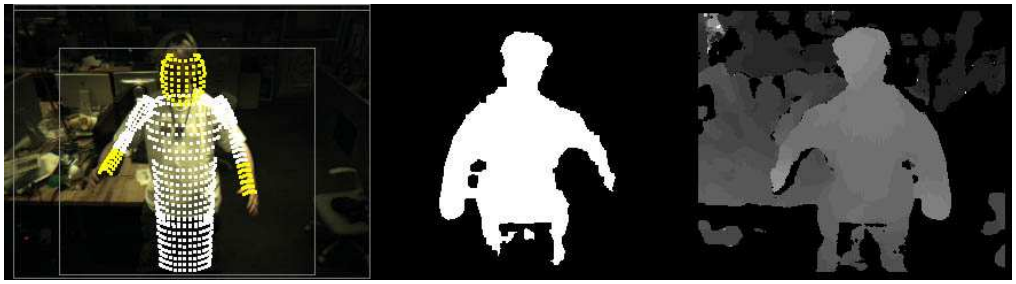


Figure 5-4: The robot currently has untethered real-time perception of both human upper arms and forearms, using the VTracker articulated body tracker.

Subsequently, for Leonardo, each of these modules was replaced. Human body tracking is performed by the VTracker[1] articulated body tracker, which uses an interval constraint propagation algorithm on stereo range data to provide a real-time position estimate of the human's torso and left and right shoulders, elbows and wrists[81] (Figure 5-4). Speech recognition and parsing is performed by CMU Sphinx 4, using grammars developed in our laboratory[68]. I have developed my own basic implementations for the spatial database and intermodule communications, though compatibility with the SES has been retained.

5.3.1 Gesture Extraction

Extraction of static nucleus deictic gestures is straightforward but varies with the visual tracking system. The Robonaut vision system tracks only the human forearm, so the principal axis of the forearm from elbow to wrist is used as the pointing vector. The VTracker system tracks both forearm and upper arm, so I use the vector between the shoulder and the wrist, which actually provides a better estimate of the distant pointing location than the forearm alone for typical human pointing gestures.

To extract the preparation phase of deictic gestures, or those with a dynamic nucleus, the time series of the position of the end effector is used instead. Principal component analysis is performed on a moving buffer of position values prior to the speech trigger. The first principal component is used as the orientation vector of the gesture. This vector is then linearly best fit to the collection of position values to determine its translational position and direction. This method has a lower accuracy than the static case, but is typically used in situations when the problem is well constrained by the spatial arrangement of the objects or the hierarchical context, as users tend to point more carefully when the situation is ambiguous.

5.3.2 Object Matching

The spatial reasoning system performs geometric matching between extracted deictic gestures and lists of candidate objects returned by the spatial database. The spatial database stores an object by its name and six floating point numbers representing its position and orientation in the robot's

[1]Developed and kindly provided for our use by Dr David Demirdjian of the Vision Interfaces Group at the MIT Computer Science and Artificial Intelligence Laboratory.

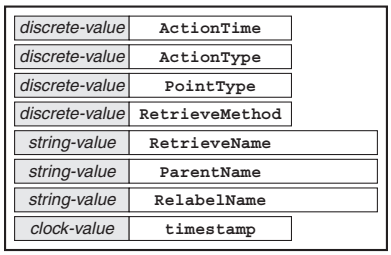


Figure 5-5: Contents of the speech message data structure sent to the spatial reasoning system to control and coördinate its activities.

coördinate frame of reference. Currently the spatial database returns all objects satisfying a query by full or partial object name, although I have designed for future support of querying by object hierarchy or by spatial partition based on the set of gestures involved in the particular circumstance.

The returned set of candidate objects is compared against the set of gestures using a linear best fit. Presently the size of the gesture queue is limited to 12 gestures, which allows an exhaustive search of the matching space to be performed to avoid encountering local minima. Clearly the accuracy of the system increases as the number of simultaneous matches is performed, but I do not envisage any tasks for Leonardo that require simultaneous sequencing of more than this number of identical objects.

Due to limitations in the spatial databases used so far, which do not store extent information, objects are currently treated as point targets and gestures as vectors. When a parent object is used as a spatial constraint, the horizontal axes of the parent object, rotated to match the pose of the object, are extrapolated to an intersection plane that reduces the gesture vectors to points from which linear distances can be calculated. When no parent object has been defined, the orthogonal distance from the candidate objects to the gesture vectors is used.

5.3.3 Deictic Grammar Parsing

The deictic grammar given in Section 5.2.4 represents a system model, of which my implementation is merely one particular realization. In my implementation the input to the grammar parser consists of a sequence of activated speech tags produced by the speech recognition system as a result of its parse of the human's speech. The structure of the tag is shown in Figure 5-5.

There are four discrete-valued fields. The *ActionTime* field indicates whether the tag represents a tag to be enqueued or executed immediately. The *ActionType* field defines the action context of the tag (and its corresponding gesture), such as a deictic pick or place action. The *PointType* field can be used to force the gesture extraction mechanism to limit itself to a static or dynamic interpretation of the arm state, if desired. The *RetrieveMethod* field informs the spatial reasoning system which parameters to use in its retrieval of object candidates from the spatial database. The string fields *RetrieveName*, *ParentName* and *RelabelName* contain the values for spatial database retrieval and object referent reposting, if applicable. Finally, a timestamp assists in maintaining synchronizations between the various modules, which do not necessarily share a common clock.

These incoming speech tags are processed by using their contents to trigger changes in the state of the spatial reasoning system that simulate the behavior of the deictic grammar model. A state

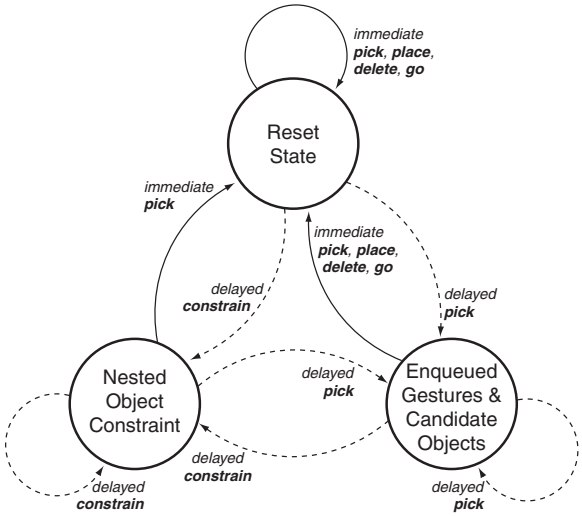


Figure 5-6: General state model of the spatial reasoning system. Edge transitions are caused by incoming speech messages, and may have communications with the spatial database associated with them. Solid edges represent transitions in which new object information may be posted to the spatial database (e.g. object relabeling or coördinate transformations). Dashed edges represent transitions in which object information may be read but not posted.

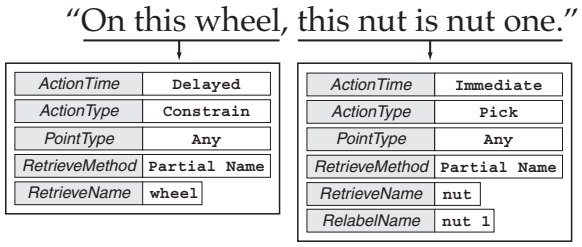


Figure 5-7: The example utterance as parsed into speech tag messages, each of which triggers an object reference incorporating a deictic gesture. The first determines the nested object constraint and the second matches the ultimate object referent and attaches its new label.

diagram of the reasoning system is shown in Figure 5-6, and an example tag sequence corresponding to the example parse of Figure 5-2 is shown in Figure 5-7. The tag structure is easily extended to more complex grammars and modules (e.g. affirmation/negation of pointing results, more complex database lookups) with corresponding alterations to the state model. Properly formed input from the speech system is assumed, and tag sequences resulting in error conditions are simply discarded.

5.4 Results and Evaluation

The initial version of this system was used as part of NASA's humanoid robot Robonaut for its Demo 2 for the DARPA MARS program, a report on which can be found in [27]. In this task, the human instructed the robot to tighten the four nuts on an ordinary automobile wheel in a specific sequence by using deictic gesture and speech to constrain the robot's attention to the wheel object

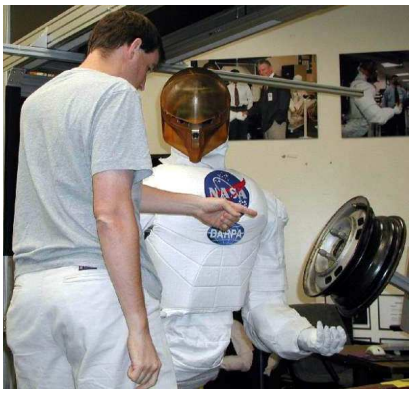


Figure 5-8: Execution of the pointing and labeling component of the Robonaut nut-tightening task. The four nuts are indicated by the human in order to label them with their sequence numbers, and the gestures then matched to the correct object referents simultaneously.

and then similarly pointing and labeling the nuts in order (Figure 5-8). The discrimination distance between the nuts was roughly on the order of the size of the nuts themselves (approximately 3cm).

Using multiple simultaneous gesture-object matching, the deictic spatial reference system was able to robustly resolve the object referents correctly in the presence of tracking and human pointing error. In the Robonaut task, an additional constraint was included: in the event that the spatial database returned the same number of candidate object referents as gestures, the system was to assume a unique one-to-one match. In such cases the system was also able to detect single duplicate labeling errors.

The current version of the system, extended for mixed-reality workspace support, is intended for use in collaborative tasks with our humanoid robot Leonardo, replacing a less sophisticated earlier deictic object reference system that operated in two spatial dimensions. The system is expected to support significantly more complex natural human-robot interactions involving objects in the world.

Other than the successful empirical evaluation of the system in use, quantitative experimental analysis of the system's performance has not been undertaken. There are two principal reasons for this. First, the major contribution represented by this work is in the conceptual framework of a deictic grammar for spatial object reference, in which both gestural and speech components are treated as linguistic elements, rather than in the accuracy or efficiency of a particular implementation of it. The implementation that was produced was successful under real time interaction conditions, validating the underlying conceptual basis.

Second, it is not clear what would be an appropriate control scenario for quantitative experimentation with the system. Quantitative measurement of the accuracy of human pointing gestures under interaction conditions that are not artificially constrained is difficult if not impossible. Perhaps a laser could be attached to human subjects' hands and then, after they had been asked to point at precise locations, the linear distance between the laser spot and the target point measured — but this immediately brings up questions of whether the angular error is more important than the linear error at the destination, or whether the laser is attached to the most appropriate place and aligned correctly, or how to avoid subjects using the laser spot as a visual feedback reference. Other studies that have required humans to point out directions or locations have used pointers

attached to external dials[77], but this is clearly straying from the natural deictic interaction in which we are interested. In any case, such experiments would only serve to demonstrate what is already trivially easy to observe: human pointing gestures are not precisely accurate but depend heavily on external context. Thus the results of any quantitative comparison of this system with a system relying purely on human pointing accuracy would produce numbers that were essentially meaningless. I therefore feel confident in declaring this system to be superior to any system that relies solely on the accuracy of the gestural component of deictic reference.

Chapter 6

Non-Verbal Communication

When humans interact with other humans, they use a variety of implicit mechanisms to share information about their own state and the state of the interaction. Expressed over the channel of the physical body, these mechanisms are collectively known as non-verbal communication or "body language". It has not been proven that humans respond in precisely the same way to the body language of a humanoid robot as they do to that of a human. Nor have the specific requirements that the robot must meet in order to ensure such a response been empirically established. It has, however, been shown that humans will apply a social model to a sociable robot[40], and will in many cases approach interactions with electronic media holding a set of preconceived expectations based on their experiences of interactions with other humans[245]. If these social equivalences extend to the interpretation of human-like body language displayed by a robot, it is likely that there will be corresponding benefits associated with enabling robots to successfully communicate in this fashion. For a robot whose principal function is interaction with humans, we identify three such potential benefits.[1]

First is the practical benefit of increasing the data bandwidth available for the "situational awareness" of the human, by transmitting more information without adding additional load to existing communication mechanisms. If it is assumed that it is beneficial for the human to be aware of the internal state of the robot, yet there are cases in which it is detrimental for the robot to interrupt other activities (e.g. dialog) in order to convey this information, an additional simultaneous data channel is called for. Non-verbal communication is an example of such a channel, and provided that the cues are implemented according to cultural norms and are convincingly expressible by the robot, adds the advantage of requiring no additional training for the human to interpret.

The second benefit is the forestalling of miscommunication within the expanded available data bandwidth. The problem raised if humans do indeed have automatic and unconscious expectations of receiving state information through bodily signals, is that certain states are represented by null signals, and humans interacting with a robot that does not communicate non-verbally (or which does so intermittently or ineffectively) may misconstrue lack of communication as deliberate communication of such a state. For example, failing to respond to personal verbal communications

[1]The work in this chapter is unique in the thesis in that it was performed under the supervision and with the contribution of Dr Ronald C. Arkin. I therefore use the word "we" throughout the chapter to reflect this.

Figure 6-1: Sony QRIO, an autonomous humanoid robot designed for entertainment and interaction with humans, shown here in its standard posture.

with attentive signals, such as eye contact, can communicate coldness or indifference. If a humanoid robot is equipped with those sorts of emotions, it is imperative that we try to ensure that such "false positive" communications are avoided to the fullest extent possible.

The third potential benefit is an increased probability that humans will be able to form bonds with the robot that are analogous to those formed with other humans — for example, affection and trust. We believe that the development of such relations requires that the robot appear "natural" — that its actions can be seen as plausible in the context of the internal and external situations in which they occur. In other words, if a person collaborating with the robot can "at a glance" gain a perspective of not just what the robot is doing but why it is doing it, and what it is likely to do next, we think that he or she will be more likely to apply emotionally significant models to the robot. A principal theory concerning how humans come to be so skilled at modeling other minds is Simulation Theory, which states that humans model the motivations and goals of an observed agent by using their own cognitive structures to mentally simulate the situation of the observee [75, 111, 126]. This suggests that it is likely that the more the observable behavior of the robot displays its internal state by referencing the behaviors the human has been conditioned to recognize — the more it "acts like" a human in its "display behaviors" — the more accurate the human's mental simulation of the robot can become.

With these benefits in mind as ultimate goals, we hereby report on activities towards the more immediate goal of realizing the display behaviors themselves, under three specific constraints. First, such displays should not restrict the successful execution of "instrumental behaviors", the tasks the robot is primarily required to perform. Second, the application of body language should be tightly controlled to avoid confusing the human — it must be expressed when appropriate, and suppressed when not. Third, non-verbal communication in humans is subtle and complex; the robot must similarly be able to use the technique to represent a rich meshing of emotions, motivations and memories. To satisfy these requirements, we have developed the concept of behavioral "overlays" for incorporating non-verbal communication displays into pre-existing robot behaviors [49]. First, overlays provide a practical mechanism for modifying the robot's pre-existing activities "on-the-fly" with expressive information rather than requiring the design of specific new activities to incorporate it. Second, overlays permit the presence or absence of body language, and the degree to which it is expressed, to be controlled independent of the underlying activity. Third, the overlay system

can be driven by an arbitrarily detailed model of these driving forces without forcing this model to be directly programmed into every underlying behavior, allowing even simple activities to become more nuanced and engaging.

A brief summary of the contributions of this chapter follows:

1. This work broadens and reappraises the use of bodily expressiveness in humanoid robots, particularly in the form of proxemics, which has hitherto been only minimally considered due to safety considerations and the relative scarcity of mobile humanoid platforms.
2. This work introduces the concept of a behavioral overlay for non-verbal communication that both encodes state information into the physical output of ordinary behaviors without requiring modification to the behaviors themselves, and increases non-verbal bandwidth by injecting additional communicative behaviors in the absence of physical resource conflicts.
3. This chapter further develops the behavioral communications overlay concept into a general model suitable for application to other robotic platforms and information sources.
4. In contrast with much prior work, the research described here provides more depth to the information that is communicated non-verbally, giving the robot the capability of presenting its internal state as interpreted via its own individuality and interpersonal memory rather than simply an instantaneous emotional snapshot.
5. Similarly, while expressive techniques such as facial feature poses are now frequently used in robots to communicate internal state and engage the human, this work makes progress towards the use of bodily expression in a goal-directed fashion.
6. Finally, this work presents a functioning implementation of a behavioral overlay system on a real robot, the Sony QRIO (Figure 6-1), including the design of data structures to represent the robot's individual responsiveness to specific humans and to its own internal model.

6.1 Proxemics and Body Language

Behavioral researchers have comprehensively enumerated and categorized various forms of non-verbal communication in humans and animals. In considering non-verbal communication for a humanoid robot, we have primarily focused on the management of spatial relationships and personal space (proxemics) and on bodily postures and movements that convey meaning (kinesics). The latter class will be more loosely referred to as "body language", to underscore the fact that while many kinesic gestures can convey meaning in their own right, perhaps the majority of kinesic contributions to non-verbal communication occurs in a paralinguistic capacity, as an enhancement of concurrent verbal dialog[86]. The use of this term is not intended, however, to imply that these postures and movements form a true language with discrete rules and grammars; but as Machotka and Spiegel point out, they convey coded messages that humans can interpret[177]. Taken together, proxemics and body language can reflect some or all of the type of interaction, the relations between participants, the internal states of the participants and the state of the interaction.

6.1.1 Proxemics

Hall, pioneer of the field of proxemics, identified a number of factors that could be used to analyze the usage of interpersonal space in human-human interactions[120]. State descriptors include the

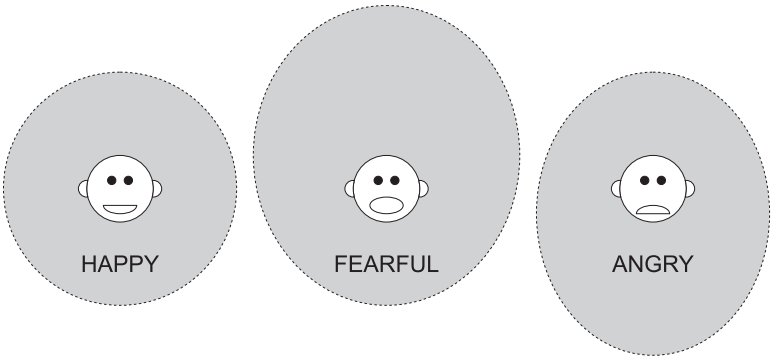


Figure 6-2: Illustration of how an individual's 'personal' space zone may vary in size and shape according to emotion. During fear, the space an individual considers his own might expand, with greater expansion occurring to his rear as he avoids potential threats that he cannot see. During anger, the space an individual considers her own might expand to a greater extent to her front as she directs her confrontational attention to known presences.

potential for the participants to touch, smell and feel the body heat of one another, and the visual appearance one another's face at a particular distance (focus, distortion, domination of visual field). The reactions of individuals to particular proxemic situations were documented according to various codes, monitoring aspects such as the amount and type of visual and physical contact, and whether or not the body posture of the subjects was encouraging ("sociopetal") or discouraging ("sociofugal") of such contact.

An informal classification was used to divide the continuous space of interpersonal distance into four general zones according to these state descriptors. In order of increasing distance, these are "Intimate", "Personal", "Socio-Consultive" and "Public". Human usage of these spaces in various relationships and situations has been observed and summarized[299], and can be used to inform the construction of a robotic system that follows similar guidelines (subject to variations in cultural norms).

Spatial separation management therefore has practical effects in terms of the potential for sensing and physical contact, and emotional effects in terms of the comfort of the participants with a particular spatial arrangement. What constitutes an appropriate arrangement depends on the nature of the interaction (what kinds of sensing and contact are necessary, and what kinds of emotional states are desired for it), the relationship between the participants (an appropriate distance for a married couple may be different than that between business associates engaged in the same activity), and the current emotional states of the participants (the preceding factors being equal, the shape and size of an individual's ideal personal "envelope" can exhibit significant variation based on his or her feelings at the time, as shown in Figure 6-2).

To ensure that a robot displays appropriate usage of and respect for personal space, and to allow it to take actions to manipulate it in ways that the human can understand and infer from them the underlying reasoning, requires consideration of all of these factors. In addition, the size of the robot (which is not limited to the range fixed by human biology) may have to be taken into account when considering the proxemics that a human might be likely to find natural or comfortable. See Table 6.1 for a comparison of several proxemic factors in the case of adult humans and QRIO, and Figure 6-3 for the general proxemic zones that were selected for QRIO.

Proxemic Factor	Human	QRIO
Kinesthetic potential (arms only)	60–75cm	20cm
Kinesthetic potential (arms plus torso)	90–120cm	25–35cm
Minimum face recognition distance	$<$ 5cm	20cm

Table 6.1: A comparison of select proxemic factors that differ between adult humans and QRIO (equipped with standard optics).

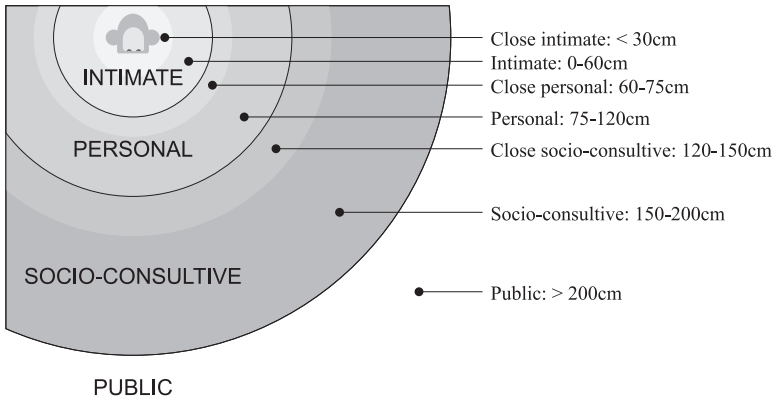


Figure 6-3: QRIO's proxemic zones in this implementation, selected as a balance between the zones for an adult human and those computed using QRIO's relevant proxemic factors. The demarcation distances between zones represent the midpoint of a fuzzy threshold function, rather than a 'hard' cutoff.

There has been little exploration of the use of proxemics in human-robot interaction to date. The reasons for this are perhaps mostly pragmatic in nature. Robotic manipulators, including humanoid upper torsos, can be dangerous to humans and in most cases are not recommended for interactions within distances at which physical contact is possible. In addition, humanoid robots with full mobility are still relatively rare, and those with legged locomotion (complete humanoids) even more so, precluding such investigations. However, some related work exists.

The mobile robot 'Chaser' by Yamasaki and Anzai focused on one of the practical effects of interpersonal distance by attempting to situate itself at a distance from the human that was ideal for sensor operation, in this case the collection of speech audio[307]. This work demonstrated that awareness of personal distance considerations could be acted upon to improve speech recognition performance.

In a similar vein, Kanda et al. performed a human-robot interaction field study in which the robot distinguished concurrently present humans as either participants or observers based on their proxemic distance[144]. A single fixed distance threshold was used for the classification, however, and it was left up to the human participants to maintain the appropriate proxemics.

Likhachev and Arkin explored the notion of "comfort zones" for a mobile robot, using attachment theory to inform an emotional model that related the robot's comfort to its spatial distance from an object of attachment[172]. The results of this work showed that the robot's exploration behavior varied according to its level of comfort; while the published work did not deal with human-robot interaction directly, useful HRI scenarios could be envisaged for cases in which the object of attachment was a human.

More recently, there have been investigations into modifying the spatial behavior of non-humanoid mobile robots in order to make people feel more at ease with the robot. Smith investigated self-adaptation of the behavior of an interactive mobile robot, including its spatial separation from the human, based on its assessment of the human's comfort[273]. In this case the goal was for the robot to automatically learn the personal space preferences of individual humans, rather than to use spatial separation as a general form of non-verbal communication. However some of the informal human subject experiments reported are instructive as to the value of taking proxemics into account in HRI, particularly the case in which misrecognition of a human discomfort response as a comfort response leads to a positive feedback loop that results in distressing behavior on the part of the robot.

Similar efforts have used considerations of humans' personal space to affect robot navigation. Nakauchi and Simmons developed a robot whose goal was to queue up to register for a conference along with human participants[210]. The robot thus needed to determine how to move to positions that appropriately matched human queuing behavior. Althaus et al. describe a system developed to allow a robot to approach a group of people engaged in a discussion, enter the group by assuming a spatially appropriate position, and then leave and continue its navigation[8]. Christensen and Pacchierotti use proxemics to inform a control strategy for the avoidance behavior exhibited by a mobile robot when forced by a constraining passageway to navigate in close proximity to humans[62]. Pacchierotti et al. then report positive results from a pilot user study, in which subjects preferred the condition in which the robot moved fastest and signaled the most deference to the humans (by moving out of the way earliest and keeping the greatest distance away), though the subjects were all familiar and comfortable with robots[225]. While informed by proxemics theory, these efforts focus on control techniques for applying social appropriateness to navigation activity, rather than using utilizing proxemic behavior as part of a non-verbal communication suite.

Another recent experiment, by te Boekhorst et al., gave some consideration to potential effects of the distance between children and a non-humanoid robot on the children's attention to a "pass the parcel" game[281]. No significant effects were recorded, however the authors admit that the data analysis was complicated by violations of the assumptions underlying the statistical tests and therefore we believe these results should not be considered conclusive. Data from the same series of experiments was used to point out that the children's initial approach distances were socially appropriate according to human proxemics theory[296]. A follow-up experiment was conducted using the same non-humanoid robot to investigate the approach distances that adults preferred when interacting with the robot[297]. A majority, 60%, positioned themselves at a distance compatible with human proxemics theory, whereas the remainder, a significant minority of 40%, assumed positions significantly closer. While these results are generally encouraging in their empirical support of the validity of human-human proxemic theory to human-robot interactions, caution should be observed in extrapolating the results in either direction due to the non-humanoid appearance of the robot.

More recently than the work in this chapter was performed, there has been interest shown in the use of proxemics and non-verbal communication by the robotic search and rescue community. In a conference poster Bethel and Murphy proposed a set of guidelines for affect expression by appearance-constrained (i.e. non-humanoid) rescue robots based on their proxemic zone with respect to a human[22]. The goal of this work was once again to improve the comfort level of humans through socially aware behavior.

While not involving robots per se, some virtual environment (VE) researchers have examined reactions to proxemic considerations between humans and humanoid characters within immersive VEs. Bailenson et al., for example, demonstrated that people exhibited similar personal spatial behavior towards virtual humans as they would towards real humans, and this effect was increased the more the virtual human was believed to be the avatar of a real human rather than an agent controlled by the computer[16]. However, they also encountered the interesting result that subjects exhibited more pronounced avoidance behavior when proxemic boundaries were violated by an agent rather than an avatar; the authors theorize that this is due to the subjects attributing more rationality and awareness of social spatial behavior to a human-driven avatar than a computer-controlled agent, thus "trusting" that the avatar would not walk into their virtual bodies, whereas an agent might be more likely to do so. This may present a lesson for HRI designers: the precise fact that an autonomous robot is known to be under computer control may make socially communicative proxemic awareness (as opposed to simple collision avoidance) particularly important for robots intended to operate in close proximity with humans.

6.1.2 Body Language

Body language is the set of communicative body motions, or kinesic behaviors, including those that are a reflection of, or are intended to have an influence on, the proxemics of an interaction. Knapp identifies five basic categories:

1. Emblems, which have specific linguistic meaning and are what is most commonly meant by the term 'gestures';
2. Illustrators, which provide emphasis to concurrent speech;
3. Affect Displays, more commonly known as facial expressions and used to represent emotional states;
4. Regulators, which are used to influence conversational turn-taking; and
5. Adaptors, which are behavioral fragments that convey implicit information without being tied to dialog[153].

Dittmann further categorizes body language into discrete and continuous (persistent) actions, with discrete actions further partitioned into categorical (always performed in essentially the same way) and non-categorical[86]. Body posture itself is considered to be a kinesic behavior, inasmuch as motion is required to modify it, and because they can be modulated by attitude[153].

The kinds of body language displays that can be realized on a particular robot of course depend on the mechanical design of the robot itself, and these categorizations of human body language are not necessarily of principal usefulness to HRI designers other than sometimes suggesting implementational details (e.g. the necessity of precisely aligning illustrators with spoken dialog). However, it is useful to examine the broad range of expression that is detailed within these classifications, in order to select those appearing to have the most utility for robotic applications. For example, classified within these taxonomies are bodily motions that can communicate explicit symbolic concepts, deictic spatial references (e.g. pointing), emotional states and desires, likes and dislikes, social status, engagement and boredom. As a result there has been significant ongoing robotics research overlapping with all of the areas thus referenced. For a comprehensive survey of socially interactive robotic research in general, incorporating many of these aspects, see [99].

The use of emblematic gestures for communication has been widely used on robotic platforms and in the field of animatronics; examples are the MIT Media Lab's 'Leonardo', an expressive humanoid which currently uses emblematic gesture as its only form of symbolic communication and also incorporates kinesic adaptors in the form of blended natural idle motions[43], and Waseda University's WE-4RII 'emotion expression humanoid robot' which was also designed to adopt expressive body postures[312].

Eye contact has also been used as a communicative enhancement to emblematic gesture recognition on robots. Miyauchi et al. took advantage of the temporal correlation of eye contact with intentional gestural communication in order to assist in the disambiguation of minimal gestures [199]. In this case the presence or absence of eye contact was used as an activation "switch" for the gesture recognizers, rather than as a communication channel to be modulated.

Comprehensive communicative gesture mechanisms have also been incorporated into animated humanoid conversational agents and VE avatars. Kopp and Wachsmuth used a hierarchical kinesic model to generate complex symbolic gestures from gesture phrases, later interleaving them tightly with concurrent speech[157, 158]. Guye-Vuilleme et al. provided collaborative VE users with the means to manually display a variety of non-verbal bodily expressions on their avatars using a fixed palette of potential actions[118].

Similarly, illustrators and regulators have been used to punctuate speech and control conversational turn-taking on interactive robots and animated characters. Aoyama and Shimomura implemented contingent head pose (such as nodding) and automatic filler insertion during speech interactions with Sony QRIO[11]. Extensive work on body language for animated conversational agents has been performed at the MIT Media Lab, such as Thorisson's implementation of a multimodal dialog skill management system on an animated humanoid for face-to-face interactions [282] and Cassell and Vilhjalmsson's work on allowing human-controlled full-body avatars to exhibit communicative reactions to other avatars autonomously[59].

Robots that communicate using facial expression have also become the subject of much attention, too numerous to summarize here but beginning with well-known examples such as Kismet and the face robots developed by Hara[37, 122]. In addition to communication of emotional state, some of these robots have used affective facial expression with the aim of manipulating the human, either in terms of a desired emotional state as in the case of Kismet or in terms of increasing desired motivation to perform a collaborative task as in subsequent work on Leonardo[50].

However much of this related work either focuses directly on social communication through body language as the central research topic rather than the interoperation of non-verbal communication with concurrent instrumental behavior, or on improvements to the interaction resulting from directly integrating non-verbal communication as part of the interaction design process. When emotional models are incorporated to control aspects such as affective display, they tend to be models designed to provide a "snapshot" of the robot's emotional state (for example represented by a number of discrete categories such as the Ekman model [94]) suitable for immediate communication via the robot's facial actuators but with minimal reference to the context of the interaction. However recent research by Fridlund has strongly challenged the widely accepted notion that facial expressions are an unconscious and largely culturally invariant representation of internal emotional state, arguing instead that they are very deliberate communications that are heavily influenced by the context in which they are expressed[100]. This is a contention that may be worth keeping in mind concerning robotic body language expression in general.

Given the capabilities of the robotic platform used for this work (Sony QRIO) and the relevance of the various types of body language to the interactions envisaged for QRIO, the following aspects were chosen for specific attention:

I. Proxemics and the management of interpersonal distance, including speed of locomotion;

II. Emblematic hand and arm gestures in support of the above;

III. The rotation of the torso during interaction, which in humans reflects the desire for interaction (facing more squarely represents a sociopetal stance, whereas displaying an angular offset is a sociofugal posture) — thus known as the "sociofugal/sociopetal axis";

IV. The posture of the arms, including continuous measures (arms akimbo, defensive raising of the arms, and rotation of the forearms which appear sociopetal when rotated outwards and sociofugal when rotated inwards) and discrete postural stances (e.g. arms folded);

V. Head pose and the maintenance of eye contact;

VI. Illustrators, both pre-existing (head nodding) and newly incorporated (attentive torso leaning).

See Figure 6-15 in Section 6.6 for examples of some of these poses as displayed by QRIO.

6.2 Behavioral Overlays

The concept of a behavioral overlay for behavior-based robots can be described as a motor-level modifier that alters the resulting appearance of a particular output conformation (a motion, posture, or combination of both). The intention is to provide a simultaneous display of information, in this case non-verbal communication, through careful alteration of the motor system in such a way that the underlying behavioral activities of the robot may continue as normal. Just as the behavior schemas that make up the robot's behavioral repertoire typically need not know of the existence or method of operation of one another, behavioral overlays should be largely transparent to the behavior schema responsible for the current instrumental behavior at a given time. Preservation of this level of modularity simplifies the process of adding or learning new behaviors.

As a simplified example, consider the case of a simple 2-DOF output system: the tail of a robotic dog such as AIBO, which can be angled around horizontal and vertical axes. The tail is used extensively for non-verbal communication by dogs, particularly through various modes of tail wagging. Four examples of such communications via the tail from the AIBO motor primitives design specification are the following:

- *Tail Wagging Friendly*: Amplitude of wag large, height of tail low, speed of wag baseline slow but related to strength of emotion.
- *Tail Wagging Defensive*: Amplitude of wag small, height of tail high, speed of wag baseline fast but related to strength of emotion.
- *Submissive Posture*: In cases of low dominance/high submission, height of tail very low (between legs), no wagging motion.

- *"Imperious Walking"*: Simultaneous with locomotion in cases of high dominance/low submission; amplitude of wag small, height of tail high, speed of wag fast.

One method of implementing these communication modes would be to place them within each behavioral schema, designing these behaviors with the increased complexity of responding to the relevant emotional and instinct models directly. An alternative approach — behavioral overlays — is to allow simpler underlying behaviors to be externally modified to produce appropriate display activity. Consider the basic walking behavior $\mathbf{b}_0$ in which the dog's tail wags naturally left and right at height ϕ_0 with amplitude $\pm\theta_0$ and frequency ω_0 from default values for the walking step motion rather than the dog's emotional state. The motor state $\mathbf{M}_T$ of the tail under these conditions is thus given by:

$$\mathbf{M}_T(t) = \begin{bmatrix} \mathbf{b}_{0_x}(t) \\ \mathbf{b}_{0_y}(t) \end{bmatrix} = \begin{bmatrix} \theta_0 sin(\omega_0 t) \\ \phi_0 \end{bmatrix}$$

Now consider a behavioral overlay vector for the tail $\mathbf{o}_0 = [\alpha, \lambda, \delta]$ applied to the active behavior according to the mathematics of a multidimensional overlay coördination function Ω_T to produce the following overlaid tail motor state $\mathbf{M}_\mathbf{T}^+$:

$$\mathbf{M}_T^+(t) = \Omega_T(\mathbf{o}_0, \mathbf{b}_0(t)) = \begin{bmatrix} \alpha\theta_0 sin(\lambda\omega_0 t) \\ \phi_0 + \delta \end{bmatrix}$$

For appropriate construction of $\mathbf{o}_0$, the overlay system is now able to produce imperious walking ($\alpha \ll 1$, $\lambda \gg 1$, $\delta \gg 0$) as well as other communicative walk styles not specifically predefined (e.g. "submissive walking": $\alpha = 0$, $\delta \ll 0$), without any modification of the underlying walking behavior. Moreover, the display output will continue to reflect as much as possible the parameters of the underlying activity (in this case the walking motion) in addition to the internal state used to generate the communications overlay (e.g. dominance/submission, emotion).

However in our AIBO example so far, the overlay system is only able to communicate the dog's internal state using the tail when it is already being moved by an existing behavior (in this case walking). It may therefore be necessary to add an additional special type of behavior whose function is to keep the overlay system supplied with motor input. This type of behavior is distinguished by two characteristics: a different stimulus set than normal behaviors (either more or less, including none at all); and its output is treated differently by the overlay system (which may at times choose to ignore it entirely). We refer to such behaviors as "idler" behaviors. In this case, consider idler behavior $\mathbf{b}_1$ which simply attempts to continuously wag the tail some amount in order to provide the overlay system with input to be accentuated or suppressed:

$$\mathbf{b}_1(t) = \begin{bmatrix} \theta_1 sin(\omega_1 t) \\ \phi_1 \end{bmatrix}$$

This behavior competes for action selection with the regular "environmental" behaviors as normal, and when active is overlaid by Ω_T in the same fashion. Thus the overlay system with the addition of one extremely basic behavior is able to achieve all of the four tail communications

displays originally specified above, including variations in degree and combination, by appropriate selection of overlay components based on the robot's internal state. For active behavior $\mathbf{b}_i$ and overlay $\mathbf{o}_j$, the tail activity produced is:

$$\mathbf{M}_T^+(t) = \Omega_T(\mathbf{o}_j, \mathbf{b}_i(t))$$

However, the addition of specialized "idler" behaviors provides additional opportunities for manipulation of the robot's display activity, as these behaviors can be designed to be aware of and communicate with the overlay system — for example, to enable the triggering of emblematic gestures. If the robot's normal behaviors are subject at a given time to the stimulus vector $[\mathbf{S}]$, the idler behaviors can be thought of as responding to an expanded stimulus vector $[\mathbf{S}, \Psi]$ where Ψ is the vector of feedback stimuli from the overlay system. For instance, AIBO's tail idler behavior upon receiving a feedback stimulus ψ_p might interrupt its wagging action to trace out a predefined shape P with the tail tip:

$$\mathbf{b}_1(\psi_p, t) = \begin{bmatrix} P_x(t) \\ P_y(t) \end{bmatrix}$$

In general, then, let us say that for a collection of active (i.e. having passed action selection) environmental behaviors $\mathbf{B}$ and idler behaviors $\mathbf{I}$, and an overlay vector $\mathbf{O}$, the overlaid motor state $\mathbf{M}^+$ is given according to the model:

$$\mathbf{M}^+ = \Omega(\mathbf{O}, [\mathbf{B}(\mathbf{S}), \mathbf{I}([\mathbf{S}, \Psi]))$$

This model is represented graphically in Figure 6-4. In the idealized modular case, the environmental behaviors need neither communicate directly with nor even be aware of the existence of the behavioral overlay system. For practical purposes, however, a coarse level of influence by the environmental behaviors on the overlay system is required, because a complete determination a priori of whether or not motor modification will interfere with a particular activity is very difficult to make. This level of influence has been accounted for in the model, and the input to Ω from $\mathbf{B}$ and $\mathbf{I}$ as shown incorporates any necessary communication above and beyond the motor commands themselves.

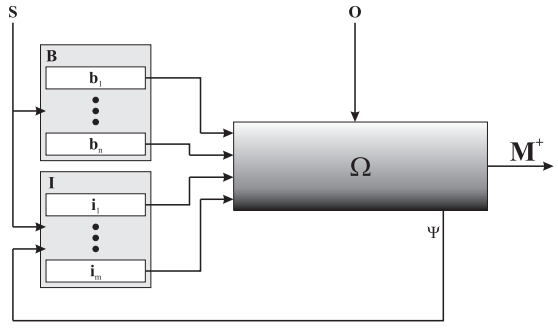


Figure 6-4: The behavioral overlay model, shown overlaying active environmental behaviors $\mathbf{b}_{1..n}$ and idler behaviors $\mathbf{i}_{1..m}$ after action selection has already been performed.

Behavioral overlays as implemented in the research described in this thesis include such facilities for schemas to communicate with the overlay system when necessary. Schemas may, if desired, protect themselves against modification in cases in which interference is likely to cause failure of the behavior (such as a task, like fine manipulation, that would similarly require intense concentration and suppression of the non-verbal communication channel when performed by a human). Care should, of course, be taken to use this facility sparingly, in order to avoid the inadvertent sending of non-verbal null signals during activities that should not require such concentration, or for which the consequences of failure are not excessively undesirable.

Schemas may also make recommendations to the overlay system that assist it in setting the envelope of potential overlays (such as reporting the characteristic proxemics of an interaction; e.g. whether a speaking behavior takes place in the context of an intimate conversation or a public address). In general, however, knowledge of and communication with the overlay system is not a requirement for execution of a behavior. As QRIO's intentional mechanism is refined to better facilitate high-level behavioral control, the intention system may also communicate with the behavioral overlay system directly, preserving the modularity of the individual behaviors.

Related work of most relevance to this concept is the general body of research related to motion parameterization. The essential purpose of this class of techniques is to describe bodily motions in terms of parameters other than their basic joint-angle time series. Ideally, the new parameters should capture essential qualities of the motion (such as its overall appearance) in such a way that these qualities can be predictably modified or held constant by modifying or holding constant the appropriate parameters. This has advantages both for motion generation (classes of motions can be represented more compactly as parameter ranges rather than clusters of individual exemplars) and motion recognition (novel motions can be matched to known examples by comparison of the parameter values).

Approaches to this technique have been applied to motion generation for animated characters can differ in their principal focus. One philosophy involves creating comprehensive sets of basic motion templates that can then be used to fashion more complex motions by blending and modifying them with a smaller set of basic parameters, such as duration, amplitude and direction; this type of approach was used under the control of a scripting language to add real-time gestural activity to the animated character OLGA[21]. At the other extreme, the Badler research group at the University of Pennsylvania argued that truly lifelike motion requires the use of a large number of parameters concerned with effort and shaping, creating the animation model EMOTE based on Laban Movement Analysis; however the model is non-emotional and does not address autonomous action generation[61].

One of the most well known examples of motion parameterization that has inspired extensive attention in both the animation and robotics communities is the technique of "verbs and adverbs" proposed by Rose et al.[250]. In this method verbs are specific base actions and adverbs are collections of parameters that modify the verbs to produce functionally similar motor outputs that vary according to the specific qualities the adverbs were designed to affect.

This technique allows, for example, an animator to generate a continuous range of emotional expressions of a particular action, from say 'excited waving' to 'forlorn waving', without having to manually create every specific example separately; instead, a single base 'waving' motion would be created, and then parameter ranges that described the variation from 'excited' to 'forlorn' provide the means of automatically situating an example somewhere on that continuum. While

the essential approach is general, it is typically applied at the level of individual actions rather than overall behavioral output due to the difficulty of specifying a suitable parameterization of all possible motion.

Similarly, the technique of "morphable models" proposed by Giese and Poggio describes motion expressions in terms of pattern manifolds inside which plausible-looking motions can be synthesized and decomposed with linear parameter coefficients, according to the principles of linear superposition[105]. In their original example, locomotion gaits such as 'walking' and 'marching' were used as exemplars to define the parameter space, and from this the parameters could be reweighted in order to synthesize new gaits such as 'limping'. Furthermore, an observed gait could then be classified against the training examples using least-squares estimation in order to estimate its relationship to the known walking styles.

Not surprisingly, motion parameterization techniques such as the above have been shown significant interest by the segment of the robotics community concerned with robot programming by demonstration. Motion parameterization holds promise for the central problem that this approach attempts to solve: extrapolation from a discrete (and ideally small) set of demonstrated examples to a continuous task competency envelope; i.e., knowing what to vary to turn a known spatio-temporal sequence into one that is functionally equivalent but better represents current circumstances that were not prevailing during the original demonstration. The robotics literature in this area, even just concerning humanoid robots, is too broad to summarize, but see [233] for a representative example that uses a verbs-adverbs approach for a learned grasping task, and illustrates the depth of the problem.

Fortunately, the problem that behavioral overlays seeks to address is somewhat simpler. In the first place, the task at hand is not to transform a known action that would be unsuccessful if executed under the current circumstances into one that now achieves a successful result; rather, it is to make modifications to certain bodily postures during known successful actions to a degree that communicates information without causing those successful actions to become unsuccessful. This distinction confers with it the luxury of being able to choose many parameters in advance according to well-described human posture taxonomies such as those referred to in Section 6.1, allowing algorithmic attention to thus be concentrated on the appropriate quantity and combination of their application. Furthermore, such a situation, in which the outcome even of doing nothing at all is at least the success of the underlying behavior, has the added advantage that unused bodily resources can be employed in overlay service with a reasonable level of assuredness that they will not cause the behavior to fail.

Secondly, the motion classification task, where it exists at all, is not the complex problem of parameterizing monolithic observed output motion-posture combinations, but simply to attempt to ensure that such conformations, when classified by the human observer's built-in parameterization function, will be classified correctly. In a sense, behavioral overlays start with motor descriptions that have already been parameterized — into the instrumental behavior itself and the overlay information — and the task of the overlay function is to maintain and apply this parameterization in such a fashion that the output remains effective in substance and natural in appearance, a far less ambiguous situation than the reverse case of attempting to separate unconstrained natural behavior into its instrumental and non-instrumental aspects (e.g. 'style' from 'content').

In light of the above, and since behavioral overlays are designed with the intention of affecting all of the robot's behavioral conformations, not just ones that have been developed or exhibited

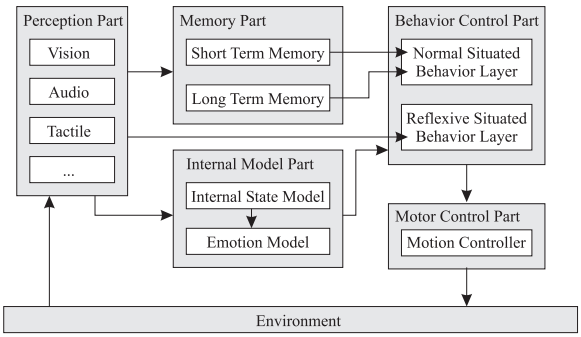


Figure 6-5: Functional units and interconnections in QRIO's standard EGO Architecture.

at the time of parameter estimation, the implementation of behavioral overlays described here has focused on two main areas. First, the development of general rules concerning the desired display behaviors and available bodily resources identified in Section 6.1 that can be applied to a wide range of the robot's activities. And second, the development of a maintenance and releasing mechanism that maps these rules to the space of emotional and other internal information that the robot will use them to express. Details of the internal data representations that provide the robot with the contextual information necessary to make these connections are given in Section 6.3, and implementation details of the overlay system itself are provided in Section 6.4.

6.3 Relationships and Attitudes

An important driving force behind proxemics and body language is the internal state of the individual. QRIO's standard EGO Architecture contains a system of internally-maintained emotions and state variables, and some of these are applicable to non-verbal communication. A brief review follows; please refer to Figure 6-5 for a block diagram of the unmodified EGO Architecture.

QRIO's emotional model (EM) contains six emotions (ANGER, DISGUST, FEAR, JOY, SADNESS, SURPRISE) plus NEUTRAL, along the lines of the Ekman proposal[94]. Currently QRIO is only able to experience one emotion from this list at a given time, though the non-verbal communication overlay system has been designed in anticipation of the potential for this to change. Emotional levels are represented by continuous-valued variables.

QRIO's internal state model (ISM) is also a system of continuous-valued variables, that are maintained within a certain range by a homeostasis mechanism. Example variables include FATIGUE, INFORMATION, VITALITY and INTERACTION. A low level of a particular state variable can be used to drive QRIO to seek objects or activities that can be expected to increase the level, and vice versa.

For more details of the QRIO emotionally grounded architecture beyond the above summary, see [258]. The remainder of this section details additions to the architecture that have been developed specifically to support non-verbal communication overlays. In the pre-existing EGO architecture, QRIO has been designed to behave differently with different individual humans by changing its EM and ISM values in response to facial identification of each human. However, these changes are instantaneous upon recognition of the human; the lack of additional factors distinguishing QRIO's

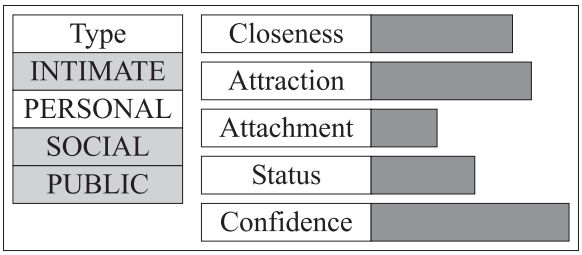


Figure 6-6: An example instantiation of a Relationship structure, showing arbitrarily selected values for the single discrete enumerated variable and each of the five continuous variables.

feelings about these specific individual humans limits the amount of variation and naturalness that can be expressed in the output behaviors.

In order for QRIO to respond to individual humans with meaningful proxemic and body language displays during personal interactions, QRIO requires a mechanism for preserving the differences between these individual partners — what we might generally refer to as a relationship. QRIO does have a long-term memory (LTM) feature; an associative memory, it is used to remember connections between people and objects in predefined contexts, such as the name of a person's favorite food. To support emotional relationships, which can then be used to influence non-verbal communication display, a data structure to extend this system has been developed.

Each human with whom QRIO is familiar is represented by a single 'Relationship' structure, with several internal variables. A diagram of the structure can be seen in Figure 6-6. The set of variables chosen for this structure have generally been selected for the practical purpose of supporting non-verbal communication rather than to mirror a particular theoretical model of interpersonal relationships, with exceptions noted below.

The discrete-valued 'Type' field represents the general nature of QRIO's relationship with this individual; it provides the global context within which the other variables are locally interpreted. Because a primary use of this structure will be the management of personal space, it has been based on delineations that match the principal proxemic zones set out by Weitz[299]. An INTIMATE relationship signifies a particularly close friendly or familial bond in which touch is accepted. A PERSONAL relationship represents most friendly relationships. A SOCIAL relationship includes most acquaintances, such as the relationship between fellow company employees. And a PUBLIC relationship is one in which QRIO may be familiar with the identity of the human but little or no social contact has occurred. It is envisaged that this value will not change frequently, but it could be learned or adapted over time (for example, a PUBLIC relationship becoming SOCIAL after repeated social contact).

All other Relationship variables are continuous-valued quantities, bounded and normalized, with some capable of negative values if a reaction similar in intensity but opposite in nature is semantically meaningful. The 'Closeness' field represents emotional closeness and could also be thought of as familiarity or even trust. The 'Attraction' field represents QRIO's desire for emotional closeness with the human. The 'Attachment' field, based on Bowlby's theory of attachment behavior[32], is a variable with direct proxemic consequences and represents whether or not the human is an attachment object for QRIO, and if so to what degree. The 'Status' field represents the relative sense of superiority or inferiority QRIO enjoys in the relationship, allowing the structure to

represent formally hierarchical relationships in addition to informal friendships and acquaintances. Finally, the Relationship structure has a 'Confidence' field which represents QRIO's assessment of how accurately the other continuous variables in the structure might represent the actual relationship; this provides a mechanism for allowing QRIO's reactions to a person to exhibit differing amounts of variability as their relationship progresses, perhaps tending to settle as QRIO gets to know them better.

In a similar vein, individual humans can exhibit markedly different output behavior under circumstances in which particular aspects of their internal states could be said to be essentially equivalent; their personalities affect the way in which their emotions and desires are interpreted and expressed. There is evidence to suggest that there may be positive outcomes to endowing humanoid robots with perceptible individual differences in the way in which they react to their internal signals and their relationships with people. Studies in social psychology and communication have repeatedly shown that people prefer to interact with people sharing similar attitudes and personalities (e.g. [26, 56]) — such behavior is known as "similarity attraction". A contrary case is made for "complementary attraction", in which people seek interactions with other people having different but complementary attitudes that have the effect of balancing their own personalities[149, 222].

These theories have been carried over into the sphere of interactions involving non-human participants. In the product design literature, Jordan discusses the "pleasurability" of products; two of the categories in which products have the potential to satisfy their users are "socio-pleasure", relating to inter-personal relationships, and "ideo-pleasure", relating to shared values[140]. And a number of human-computer interaction studies have demonstrated that humans respond to computers as social agents with personalities, with similarity attraction being the norm (e.g. [211]). Yan et al. performed experiments in which AIBO robotic dogs were programmed to simulate fixed traits of introversion or extroversion, and showed that subjects were able to correctly recognize the expressed trait; however, in this case the preferences observed indicated complementary attraction, with the authors postulating the embodiment of the robot itself as a potential factor in the reversal[308].

As a result, if a humanoid robot can best be thought as a product which seeks to attract and ultimately fulfil the desires of human users, it might be reasonable to predict that humans will be more attracted to and satisfied by robots which appear to match their own personalities and social responses. On the other hand, if a humanoid robot is instead best described as a human-like embodied agent that is already perceived as complementary to humans as a result of its differences in embodiment, it might alternatively be predicted that humans will tend to be more attracted to robots having personalities that are perceptibly different from their own. In either case, a robot possessing the means to hold such attitudes would be likely to have a general advantage in attaining acceptance from humans, and ultimately the choice of the precise nature of an individual robot could be left up to its human counterpart.

To allow QRIO to exhibit individualistic (and thus hopefully more interesting) non-verbal behavior, a system that interprets or filters certain aspects of QRIO's internal model was required. At present this system is intended only to relate directly to the robot's proxemic and body language overlays; no attempt has been made to give the robot anything approaching a complete 'personality'. As such, the data structure representing these individual traits is instead entitled an 'Attitude'. Each QRIO has one such structure; it is envisaged to have a long-to-medium-term

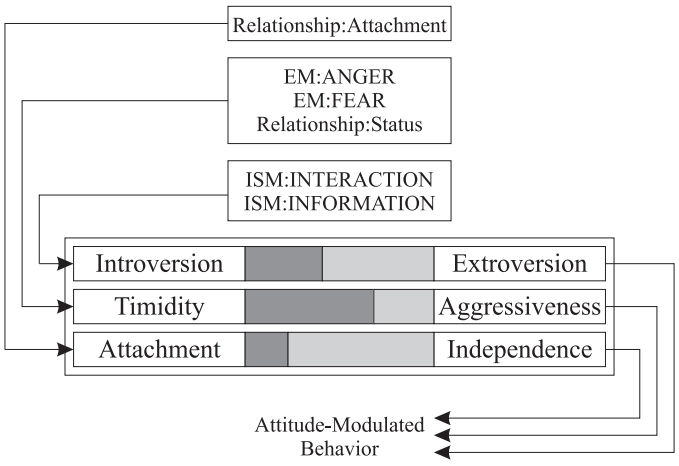


Figure 6-7: An arbitrary instantiation of an Attitude structure, showing the three opposing pairs of continuous-valued variables, and the ISM, EM and Relationship variables that each interprets.

effect, in that it is reasonable for the structure to be pre-set and not to change thereafter, but it also might be considered desirable to have the robot be able to change its nature somewhat over the course of a particularly long term interaction (even a lifetime), much as human attitudes sometimes mellow or become more extreme over time. Brief instantiation of temporary replacement Attitude (and Relationship) structures would also provide a potential mechanism for QRIO to expand its entertainment repertoire with 'acting' ability, simply by using its normal mechanisms to respond to interactions as though they featured different participants.

The Attitude structure consists of six continuous-valued, normalized variables in three opposing pairs, as illustrated in Figure 6-7. Opposing pairs are directly related in that one quantity can be computed from the other; although only three variables are therefore computationally necessary, they are specified in this way to be more intuitively grounded from the point of view of the programmer or behavior designer.

The 'Extroversion' and 'Introversion' fields are adapted directly from the Extroversion dimension of the Five-Factor Model (FFM) of personality[192], and as detailed above were used successfully in experiments with AIBO. In the case of QRIO, these values are used to affect the expression of body language and to interpret internal state desires. High values of Extroversion encourage more overt body language, whereas high Introversion results in more subtle bodily expression. Extroversion increases the effect of the ISM variable INTERACTION and decreases the effect of INFORMATION, whereas Introversion does the reverse.

The 'Aggressiveness' and 'Timidity' fields are more loosely adapted from the FFM — they can be thought of as somewhat similar to a hybrid of Agreeableness and Conscientiousness, though the exact nature is more closely tailored to the specific emotions and non-verbal display requirements of QRIO. Aggressiveness increases the effect of the EM variable ANGER and decreases the effect of FEAR, while Timidity accentuates FEAR and attenuates ANGER. High Timidity makes submissive postures more probable, while high Aggressiveness raises the likelihood of dominant postures and may negate the effect of Relationship Status.

Finally, the 'Attachment' and 'Independence' fields depart from the FFM and return to Bowlby; their only effect is proxemic, as an interpreter for the value of Relationship Attachment. While

any human can represent an attachment relationship with the robot, robots with different attitudes should be expected to respond to such relationships in different ways. Relationship Attachment is intended to have the effect of compressing the distance from the human that the robot is willing to stray; the robot's own Independence or Attachment could be used for example to alter the fall-off probabilities at the extremes of this range, or to change the 'sortie' behavior of the robot to bias it to make briefer forays away from its attachment object.

The scalar values within the Relationship and Attitude structures provide the raw material for affecting the robot's non-verbal communication (and potentially many other behaviors). These have been carefully selected in order to be able to drive the output overlays in which we are interested. However, there are many possible ways in which this material could then be interpreted and mathematically converted into the overlay signals themselves. We do not wish to argue for one particular numerical algorithm over another, because that would amount to claiming that we have quantitative answers to questions such as "how often should a robot which is 90% introverted and 65% timid lean away from a person to whom it is only 15% attracted?". We do not make such claims. Instead, we will illustrate the interpretation of these structures through two examples of general data usage models that contrast the variability of overlay generation available to a standard QRIO versus that available to one equipped with Relationships and Attitudes.

Sociofugal/sociopetal axis: We use a scalar value for the sociofugal/sociopetal axis, S, from 0.0 (the torso facing straight ahead) to 1.0 (maximum off-axis torso rotation). Since this represents QRIO's unwillingness to interact, a QRIO equipped with non-verbal communication skills but neither Relationships nor Attitudes might generate this axis based on a (possibly non-linear) function s of its ISM variable INTERACTION, I_{INT}, and its EM variable ANGER, E_{ANG}:

$$S = s(I_{INT}, E_{ANG})$$

The QRIO equipped with Relationships and Attitudes is able to apply more data towards this computation. The value of the robot's Introversion, A_{Int}, can decrease the effect of INTERACTION, resulting in inhibition of display of the sociofugal axis according to some combining function f. Conversely, the value of the robot's Aggression, A_{Agg}, can increase the effect of ANGER, enhancing the sociofugal result according to the combining function g. Furthermore, the robot may choose to take into account the relative Status, R_{Sta}, of the person with whom it is interacting, in order to politely suppress a negative display. The improved sociofugal axis generation function s^+ is thus given by:

$$S' = s^+(f(I_{INT}, A_{Int}), g(E_{ANG}, A_{Agg}), R_{Sta})$$

Proxemic distance: Even if the set of appropriate proxemic zones $\mathbf{P}$ for the type of interaction is specified by the interaction behavior itself, QRIO will need to decide on an actual scalar distance within those zones, D, to stand from the human. A standard QRIO might thus use its instantaneous EM variable FEAR, E_{FEA}, to influence whether it was prepared to choose a close value or to instead act more warily and stand back, according to another possibly non-linear function d:

$$D = d(\mathbf{P}, E_{FEA})$$

The enhanced QRIO, on the other hand, is able to make much more nuanced selections of proxemic distance. The proxemic Type field of the Relationship, R_{Typ}, allows the robot to select the most appropriate single zone from the options given in $\mathbf{P}$. The value of the robot's Timidity, A_{Tim}, increases the effect of FEAR, altering the extent to which the robot displays wariness according to a combining function u. The Closeness of the robot's relationship to the human, R_{Clo}, can also be communicated by altering the distance it keeps between them, as can its Attraction for the human, R_{Attr}. If the robot has an Attachment relationship with the human, its strength R_{Atta} can be expressed by enforcing an upper bound on D, modulated by the robot's Independence A_{Ind} according to the combining function v. The improved ideal proxemic distance generation function d^+ is thus given by:

$$D' = d^+ \left(\begin{array}{l} \mathbf{P}, R_{Typ}, u(E_{FEA}, A_{Tim}), \\ R_{Clo}, R_{Attr}, v(R_{Atta}, A_{Ind}) \end{array} \right)$$

Clearly, the addition of Relationships and Attitudes offer increased scope for variability of non-verbal communications output. More importantly, however, they provide a rich, socially grounded framework for that variability, allowing straightforward implementations to be developed that non-verbally communicate the information in a way that varies predictably with the broader social context of the interaction.

6.4 An Implementation of Behavioral Overlays

The QRIO behavioral system, termed the EGO Architecture, is a distributed object-based software architecture based on the OPEN-R modular environment originally developed for AIBO. Objects in EGO are in general associated with particular functions. There are two basic memory objects: Short-Term Memory (STM) which processes perceptual information and makes it available in processed form at a rate of 2 Hz, and Long-Term Memory (LTM) which associates information (such as face recognition results) with known individual humans. The Internal Model (IM) object manages the variables of the ISM and EM. The Motion Controller (MC) receives and executes motion commands, returning a result to the requesting module.

QRIO's actual behaviors are executed in up to three Situated Behavior Layer (SBL) objects: the Normal SBL (N-SBL) manages behaviors that execute at the 2 Hz STM update rate (homeostatic behaviors); the Reflexive SBL (R-SBL) manages behaviors that require responses faster than the N-SBL can provide, and therefore operates at a significantly higher update frequency (behaviors requiring perceptual information must communicate with perceptual modules directly rather than STM); and deliberative behavior can be realized in the Deliberative SBL (D-SBL).

Within each SBL, behaviors are organized in a tree-structured network of schemas; schemas perform minimal communication between one another, and compete for activation according to a winner-take-all mechanism based on the resource requirements of individual schemas. For more information about the EGO Architecture and the SBL system of behavior control, please see [101].

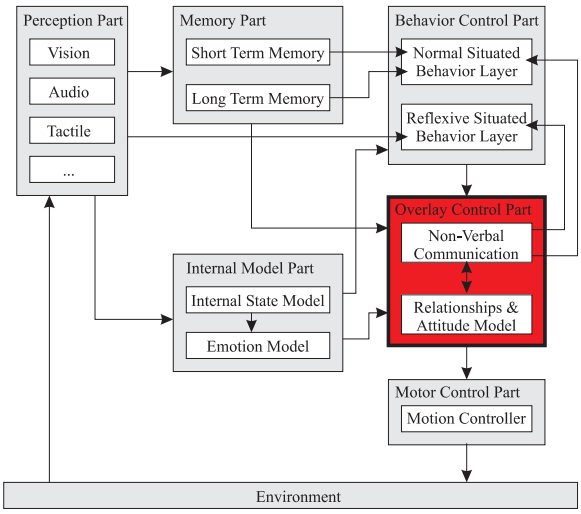


Figure 6-8: Interdependence diagram of the EGO Architecture with NVC.

6.4.1 NVC Object

Because behavioral overlays must be able to be applied to all behavior schemas, and because schemas are intended to perform minimal data sharing (so that schema trees can be easily constructed from individual schemas without having to be aware of the overall tree structure), it is not possible or desirable to completely implement an appropriate overlay system within the SBLs themselves. Instead, to implement behavioral overlays an additional, independent Non-Verbal Communication (NVC) object was added to the EGO Architecture.

The NVC object has data connections with several of the other EGO objects, but its most central function as a motor-level overlay object is to intercept and modify motor commands as they are sent to the MC object. Addition of the NVC object therefore involves reconnecting the MC output of the various SBL objects to the NVC, and then connecting the NVC object to the MC. MC responses are likewise routed through the NVC object and then back to the SBLs.

In addition to the motor connection, the NVC object maintains a connection to the IM output (for receiving ISM and EM updates, which can also be used as a 2 Hz interrupt timer), the STM target update (for acquiring information about the location of humans, used in proxemic computations) and a custom message channel to the N-SBL and R-SBL (for receiving special information about the interaction, and sending trigger messages for particular gestures and postures). See Figure 6-8 for a graphical overview of the EGO Architecture with NVC. In addition to the connections shown, the NVC object manages the behavioral overlay values themselves with reference to the Relationship and Attitude structures; Figure 6-9 has an overview of the internal workings of NVC.

6.4.2 Overlays, Resources and Timing

The data representation for the behavioral overlays themselves is basic yet flexible. They are divided according to the major resource types Head, Arms, Trunk and Legs. For each joint (or walking parameter value, in the case of the legs) within a resource type, the overlay maintains a value and

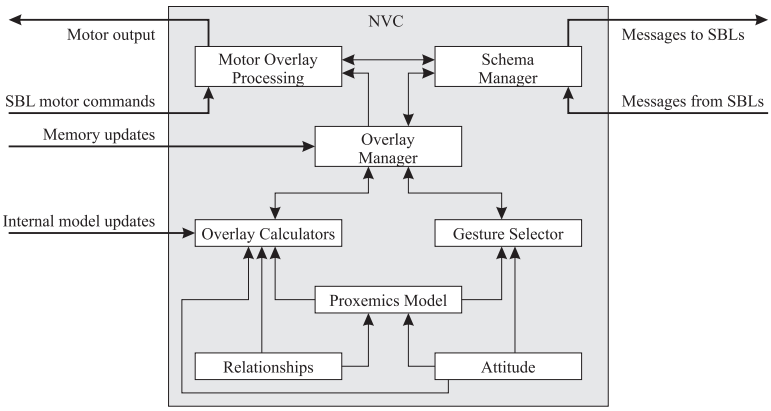


Figure 6-9: The non-verbal communication (NVC) module's conceptual internal structure and data interconnections with other EGO modules.

a flag that allows the value to be interpreted as either an absolute bias (to allow constant changes to the postural conformation) or a relative gain (to accentuate or attenuate incoming motions). In addition, each overlay category contains a time parameter for altering the speed of motion of the resource, which can also be flagged as a bias or a gain. Finally, the legs overlay contains an additional egocentric position parameter that can be used to modify the destination of the robot in the case of walking commands.

Motion commands that are routed through the NVC object consist of up to two parts: a command body, and an option parameter set. Motions that are parameterized (i.e. that have an option part) can be modified directly by the NVC object according to the current values of the overlays that the NVC object is storing. Such types of motions include direct positioning of the head, trunk and arm with explicit joint angle commands; general purpose motions that have been designed with reuse in mind, such as nodding (the parameter specifying the depth of nod); and commands that are intended to subsequently run in direct communication with perceptual systems with the SBL excluded from the decision loop, such as head tracking. Unfortunately due to the design of the MC system, unparameterized motion commands (i.e. those with just a body) cannot be altered before reaching the MC object; but they can be ignored or replaced with any other single parameterized or unparameterized motion command having the same actuator resource requirements (this possibility is not yet taken advantage of in the current implementation).

Resource management in the EGO Architecture is coarse grained and fixed; it is used for managing schema activation in the SBLs as well as just for presenting direct motion command conflicts. The resource categories in EGO are the Head, Trunk, Right Arm, Left Arm and Legs. Thus a body language motion command that wanted only to adjust the sociofugal/sociopetal axis (trunk rotate), for example, would nevertheless be forced to take control of the entire trunk, potentially blocking an instrumental behavior from executing. The NVC object implements a somewhat finer-grained resource manager by virtue of the overlay system. By being able to choose to modify the commands of instrumental behaviors directly, or not to do so, the effect is as if the resource were managed at the level of individual joints.

This raises, however, the problem of the case of time steps at which environmental behaviors do not send motion commands yet it is desired to modify the robot's body language; this situation is common. The NVC object skirts this problem by relying on a network of fast-activating idle

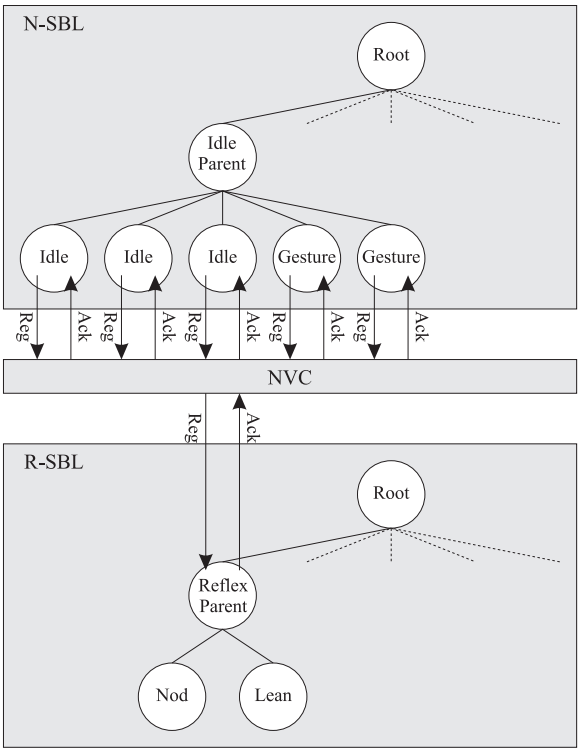


Figure 6-10: Special schemas register with the NVC object in a handshaking procedure. An acknowledgement from NVC is required because the SBLs do not know when the NVC object is ready to accept messages, so they attempt to register until successful. Within the R-SBL, the parent registers instead of the individual schemas, to preserve as closely as possible the mode of operation of the pre-existing conversational reflex system. The registration system also supports allowing schemas to deliberately change type at any time (e.g. from an idle schema to a behavioral schema) though no use of this extensibility has been made to date.

schemas residing in the N-SBL. Each idle schema is responsible for a single MC resource. On each time step, if no instrumental behavior claims a given resource, the appropriate idle schema sends a null command to the NVC object for potential overlaying. If an overlay is desired, the null command is modified into a genuine action and passed on to the MC; otherwise it is discarded. Since commands from idle and other special overlay schemas are thus treated differently by the NVC object than those from behavioral schemas, the NVC object must keep track of which schemas are which; this is accomplished by a handshaking procedure that occurs at startup time, illustrated graphically in Figure 6-10. Normal operation of the idle schemas is illustrated in Figure 6-11.

For practical considerations within the EGO Architecture, actions involved in non-verbal communication can be divided into four categories according to two classification axes. First is the timing requirements of the action. Some actions, such as postural shifts, are not precisely timed and are appropriately suited to the 2 Hz update rate of the N-SBL. Others, however, are highly contingent with the activity, such as nodding during dialog, and must reside in the R-SBL. Second is the resource requirements of the action. Some actions, such as gross posture, require only partial control of a resource, whereas others require its total control. Partial resource management has just been described above, and it functions identically with commands that are also synchronous and originate in the R-SBL. However there are also non-verbal communication behaviors that require

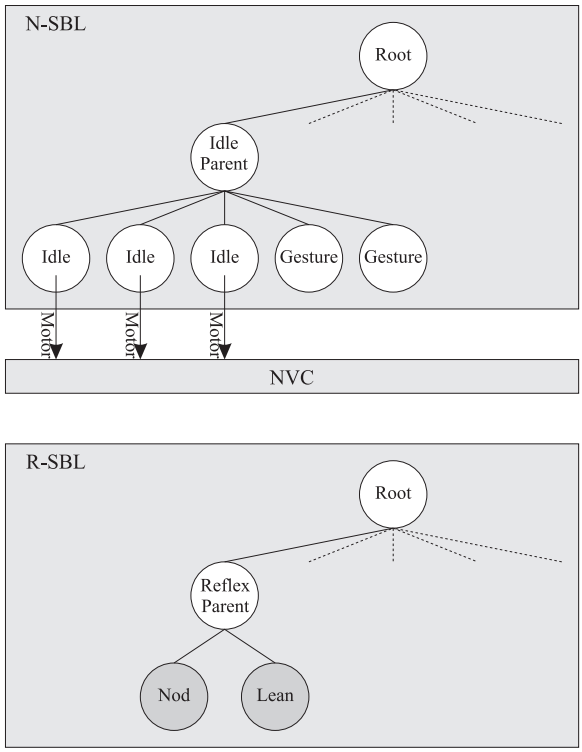


Figure 6-11: Normal idle activity of the system in the absence of gesture or reflex triggering. N-SBL idle and gesture schemas are active if resources permit. Active idle schemas send null motor commands to NVC on each time step. The dialog reflex parent is always active, but the individual dialog reflex schemas are inactive.

total resource control, such as emblematic gestures, and these are also implemented with the use of specialized "triggered" schemas: at the N-SBL level these schemas are called gesture schemas, and at the R-SBL level they are called dialog reflex schemas.

Gesture schemas reside within the same subtree of the N-SBL as the idle schemas; unlike the idle schemas, however, they do not attempt to remain active when no instrumental behavior is operating. Instead, they await gesture trigger messages from the NVC object, because the NVC object can not create motion commands directly, it can only modify them. Upon receiving such a message, a gesture schema determines if it is designed to execute the requested gesture; if so, it increases its activation level (AL), sends the motion command, and then reduces its AL again (Figure 6-12). The gesture, which may be an unparameterized motion from a predesigned set, is then executed by the MC object as usual.

Dialog reflex schemas operate in a similar but slightly different fashion. Existing research on QRIO has already resulted in a successful reactive speech interaction system that inserted contingent attentive head nods and speech filler actions into the flow of conversation[11]. It was of course desirable for the NVC system to complement, rather than compete with, this existing system. The prior system used the "intention" mechanism to modify the ALs of dialog reflex schemas, residing in the R-SBL, at appropriate points in the dialog. The NVC object manages these reflexes in almost exactly the same way.

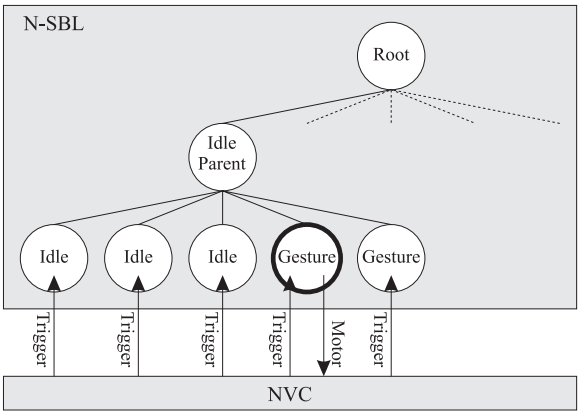


Figure 6-12: Triggering of a gesture occurs by NVC sending a trigger message to all registered N-SBL schemas. There is no distinction between idle and gesture schemas as idle schemas can include gesture-like responses such as postural shifts. Individual active schemas decode the trigger message and choose whether or not to respond with a motor command; in this typical example one gesture schema responds.

Dialog reflex schemas are grouped under a parent schema which has no resource requirements and is always active. Upon receiving a control message from the NVC object, the parent schema activates its children's monitor functions, which then choose to increase their own ALs, trigger the reflex gesture and then reduce their ALs again. Thus the only practical difference from the previous mechanism is that the reflex schemas themselves manage their own AL values, rather than the triggering object doing it for them. In addition to the existing head nodding dialog reflex, a second dialog reflex schema was added to be responsible for leaning the robot's torso in towards the speaker (if the state of the robot's interest and the Relationship deem it appropriate) in reaction to speech (Figure 6-13).

6.4.3 Execution Flow

The overall flow of execution of the NVC behavioral overlay system (Figure 6-14) is thus as follows:

- At system startup, the idle, and gesture schemas must register themselves with the NVC object to allow subresource allocation and communication. These schemas set their own AL high to allow them to send messages; when they receive an acknowledgement from NVC, they reduce their ALs to a level that will ensure that instrumental behaviors take priority (Figure 6-10).
- Also at system startup, the dialog reflex schema parent registers itself with NVC so that NVC knows where to send dialog reflex trigger messages (Figure 6-10).
- Proxemic and body language activity commences when an N-SBL schema informs NVC that an appropriate interaction is taking place. Prior to this event NVC passes all instrumental motion commands to MC unaltered and discards all idle commands (Figure 6-11). The N-SBL interaction behavior also passes the ID of the human subject of the interaction to NVC so it can look up the appropriate Relationship.

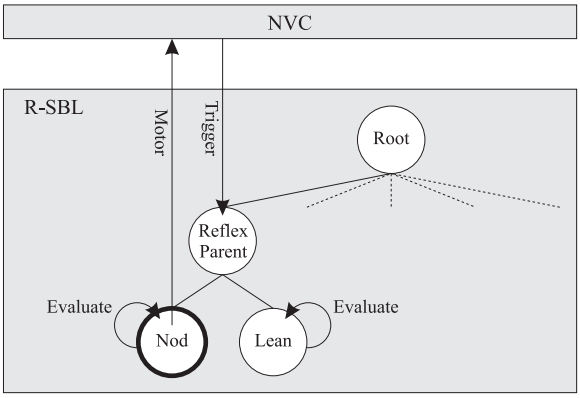


Figure 6-13: Triggering of a dialog reflex occurs by NVC sending an activation message to the active dialog reflex parent, which then signals the self-evaluation functions of its children. Children self-evaluating to active (e.g. speech is occurring) raise their own activation levels and send motor commands if required. The dialog reflex system remains active and periodically self-evaluating until receiving a deactivation message from NVC, so may either be tightly synchronized to non-causal data (e.g. pre-marked outgoing speech) or generally reactive to causal data (e.g. incoming speech).

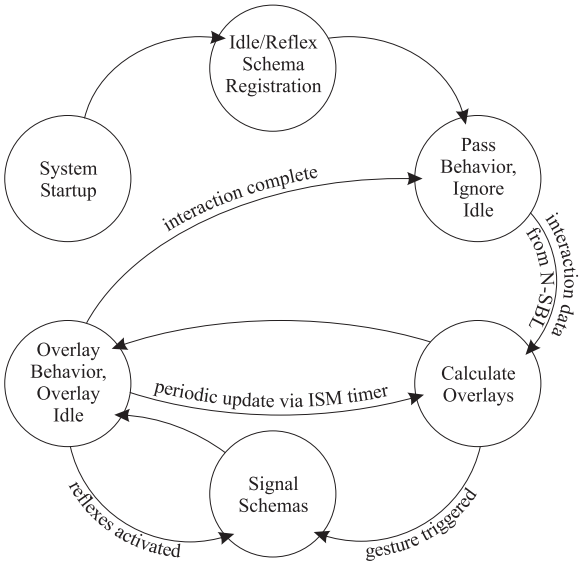


Figure 6-14: Execution flow of the NVC overlay system. Timing and synchronization issues such as responding to changes in the internal model and proxemic state (from STM) and waiting for the human to respond to an emblematic gesture are implicitly managed within the overlay calculation state (i.e. entering this state does not guarantee that any change to the overlays will in fact be made).

- NVC generates overlay values using a set of functions that take into account the IM and Attitude of the robot, the Relationship with the human and the proxemic hint of the interaction (if available). This overlay generation phase includes the robot's selection of the ideal proxemic distance, which is converted into a walking overlay if needed.

- Idle commands begin to be processed according to the overlay values. As idle commands typically need only be executed once until interrupted, the NVC object maintains a two-tiered internal subresource manager in which idle commands may hold a resource until requested by an behavior command, but completing behavior commands free the resource immediately.

- To make the robot's appearance more lifelike and less rigid, the regular ISM update is used to trigger periodic re-evaluation of the upper body overlays. When this occurs, resources held by idle commands are freed and the next incoming idle commands are allowed to execute the new overlays.

- STM updates are monitored for changes in the proxemic state caused by movement of the target human, and if necessary the overlays are regenerated to reflect the updated situation. This provides an opportunity for triggering of the robot's emblematic gestures concerned with manipulating the human into changing the proxemic situation, rather than the robot adjusting it directly with its own locomotion. A probabilistic function is executed that depends mainly on the robot's IM and Attitude and the prior response to such gestures, and according to the results an emblematic gesture of the appropriate nature and urgency (e.g. beckoning the human closer, or waving him or her away) is triggered. Trigger messages are sent to all registered idle and gesture schemas, with it up to the schemas to choose whether or not to attempt to execute the proposed action (Figure 6-12).

- Activation messages are sent from NVC to the dialog reflex schemas when dictated by the presence of dialog markers or the desire for general reactivity of the robot (Figure 6-13).

6.5 HRI Study Proposal

Having implemented non-verbal communication overlays on QRIO, it would be appropriate and desirable to scientifically evaluate their impact on QRIO's performance as an entertainment robot. I therefore designed an outline of a proposed human-robot interaction (HRI) study using QRIO. This protocol was then presented to SIDL for consideration of potentially conducting it in Japan. Unfortunately, there was not time for the study to be performed under my direct supervision due to the limited length of my internship with SIDL; and shortly after the protocol was submitted, new research priorities from the parent Sony Corporation were instated that precluded the conduct of the study in the foreseeable future. Nevertheless, it is included here as a concrete example of future work that was being planned and would ideally have occurred at some stage.

The proposed study is a single factor independent design with the two conditions 'With-NVC' and 'Without-NVC'. An estimated 20 subjects would be required, preferably gender-balanced and not having previously interacted with QRIO. Since it is likely that the novelty of interacting with a humanoid robot may have an impact on the results of this type of study, it is recommended that the study be a longitudinal study, with each subject interacting with QRIO on 3 to 4 separate

occasions. This would also offer the opportunity to vary QRIO's internal state between interactions in order to better provide the subjects with a range of observable behavior.

There have been several studies to date specifically exploring the use of non-verbal communication by robots during interactions with humans. As mentioned earlier, Yan et al. demonstrated that humans could successfully distinguish between introversion and extroversion in AIBO based on behavioral signals [308]. Breazeal et al. showed that the presence of non-verbal cues during a collaborative task with an upper torso humanoid robot led to a reduction in both the time necessary to complete the task and the number of errors occurring, and to subjects reporting a better mental model of the robot [45]. Moshkina and Arkin performed a longitudinal study of emotional expression in AIBO that linked the presence of emotion to a reduction in negative mood expressed by the participants, as well as reporting the useful caveat that across all conditions female subjects perceived the robot as more emotional than did males [204].

In contrast with the above, this study involves a humanoid robot that not only exhibits human-like postural expression but also is capable of locomotion, allowing proxemics to be included in the investigation. Thus particular attention should be paid to the spatial behavior of the subjects interacting with the robot in this study. For the rest of this section, the acronym 'NVC' refers to the non-verbal communication overlays described in this thesis, including all aspects of proxemic awareness and manipulation.

This study will attempt to investigate the following hypotheses:

Hypothesis I: NVC will cause users to respond to the robot's behavior as though it is more socially aware and "natural".

Hypothesis II: NVC will increase users' affinity for, enjoyment of, and desire for future interaction with the robot.

Hypothesis III: NVC will make users feel more strongly that the robot knows something about them as an individual.

Hypothesis IV: NVC will enhance users' ability to recognize the effect of emotions and/or personality on the robot's behavior.

Hypothesis V: NVC will affect users' choice of activities to perform with the robot.

The interaction with the robot will be a form of "command play" similar to that used in [204], aiming to make use of QRIO's design for motion and entertainment. Subjects will be informed of a limited set of commands which they can give to the robot. Some of the commands will generate purely demonstrative play activities, for example 'Dance' and 'Sports', in which QRIO displays a randomly selected motion from its existing repertoire. The other commands will generate more interactive play activities, for example 'About Me' (QRIO randomly selects from a list of known questions, then asks the user or, if already asked, repeats the answer, e.g.: "What is your favorite food?"/"Your favorite food is ...") and 'About QRIO' (QRIO asks the user to guess some information about it, and then either affirms the guess or tells the correct answer). Due to the complexities of autonomous speech interaction, the spoken activities may be implemented in 'Wizard of Oz' fashion, with a human controller ensuring correct selection of utterances.

In one of the sessions for each subject, QRIO's internal state will be emotionally negative and uninterested in interacting. In this case, when the user requests a conversational play activity, there

is a probability that QRIO will refuse to participate. In the With-NVC condition, this behavior will be signaled by a sociofugal posture and increased proxemic distance; whereas in the Without-NVC condition, only the acquiescences and refusals themselves will be observable.

Measures used to evaluate the hypotheses will be video analysis of the interaction sessions and a series of self-report questionnaires given to the subjects at the end of each session. The video will be principally used to confirm or reject Hypotheses 1 and 5. The recordings will be coded for proxemic behavior on the part of the human, examining the frequency and magnitude of spatial motions made by the human (moving or leaning closer or further from the robot), and proxemic-related gestural and postural actions. Differences in the subjects' proxemic behavior between conditions will reflect their different expectations of QRIO's social awareness based on its non-verbal communication, for Hypothesis 1. The video will also be used to note the number of times subjects request more interactive versus less interactive activities, according to Hypothesis 5.

After each session, subjects will be asked to complete a PANAS questionnaire [298] to measure their positive/negative emotionality or "mood" following interaction with QRIO, in order to help analyze Hypothesis 2. They will also be given a specific post-interaction questionnaire consisting of a series of Likert scale questions, the results of which will be used to contribute to the analyses of Hypotheses 1 through 4. Though the exact wording of the questions needs to be prepared under careful review, an example questionnaire might be:

1: It was easy to understand which action QRIO was doing.

2: It was easy to understand why QRIO did each action.

3: I feel comfortable around QRIO.

4: I enjoyed this interaction with QRIO.

5: I would like to interact with QRIO again in the future.

6: Overall, I like QRIO.

7: QRIO understood what I wanted to do.

8: QRIO knows me personally.

9: QRIO showed emotional expressions concerning this interaction.

10: QRIO encouraged me to continue the interaction.

11: I would expect another QRIO to act just like this QRIO.

A slightly different questionnaire might also be presented after the final session in order to provide additional explicit longitudinal summary data if desired.

6.6 Results and Evaluation

All of the non-verbal communication capabilities set out in Section 6.1 were implemented as part of the overlay system. To test the resulting expressive range of the robot, we equipped QRIO with a suite of sample Attitudes and Relationships. The Attitude selections comprised a timid, introverted QRIO, an aggressive, agonistic QRIO, and an "average" QRIO. The example Relationships were

crafted to represent an "old friend" (dominant attribute: high Closeness), an "enemy" (dominant attribute: low Attraction), and the "Sony President" (dominant attribute: high Status). The test scenario consisted of a simple social interaction in which QRIO notices the presence of a human in the room and attempts to engage him or her in a conversation. The interaction schema was the same across conditions, but the active Attitude and Relationship, as well as QRIO's current EM and ISM state, were concurrently communicated non-verbally. This scenario was demonstrated to laboratory members, who were able to recognize the appropriate changes in QRIO's behavior. Due to the high degree of existing familiarity of laboratory members with QRIO, however, we did not attempt to collect quantitative data from these interactions.

Due to the constraints of our arrangement with Sony to collaborate on their QRIO architecture, formal user studies of the psychological effectiveness of QRIO's non-verbal expressions (as distinct from the design of the behavioral overlay model itself, which was verified with the successful QRIO implementation) have been recommended but not yet performed. In lieu of such experiments, the results of this work are illustrated with a comparison between general interactions with QRIO (such as the example above) in the presence and absence of the overlay system. Let us specify the baseline QRIO as equipped with its standard internal system of emotions and instincts, as well as behavioral schema trees for tracking of the human's head, for augmenting dialog with illustrators such as head nodding and torso leaning, and for the conversation itself. The behavioral overlay QRIO is equipped with these same attributes plus the overlay implementation described here.

When the baseline QRIO spots the human, it may begin the conversation activity so long as it has sufficient internal desire for interaction. If so, QRIO commences tracking the human's face and responding to the human's head movements with movements of its own head to maintain eye contact, using QRIO's built-in face tracking and fixation module. When QRIO's built-in auditory analysis system detects that the human is speaking, QRIO participates with its reflexive illustrators such as head nodding, so the human can find QRIO to be responsive. However, it is up to the human to select an appropriate interpersonal distance — if the human walks to the other side of the room, or thrusts his or her face up close to QRIO's, the conversation continues as before. Similarly, the human has no evidence of QRIO's desire for interaction other than the knowledge that it was sufficiently high to allow the conversational behavior to be initiated; and no information concerning QRIO's knowledge of their relationship other than what might come up in the conversation.

Contrast this scenario with that of the behavioral overlay QRIO. When this QRIO spots the human and the conversation activity commences, QRIO immediately begins to communicate information about its internal state and the relationship it shares with the human. QRIO begins to adopt characteristic body postures reflecting aspects such as its desire for interaction and its relative status compared with that of the human — for some examples see Figure 6-15. If the human is too close or too far away, QRIO may walk to a more appropriate distance (see Figure 6-3 for specifics concerning the appropriate interaction distances selected for QRIO). Alternatively, QRIO may beckon the human closer or motion him or her away, and then give the human a chance to respond before adjusting the distance itself if he or she does not. This may occur at any time during the interaction — approach too close, for example, and QRIO will back away or gesticulate its displeasure. Repeated cases of QRIO's gestures being ignored reduces its patience for relying on the human's response.

The behavioral overlay QRIO may also respond to the human's speech with participatory illustrators. However its behavior is again more communicative: if this QRIO desires interaction

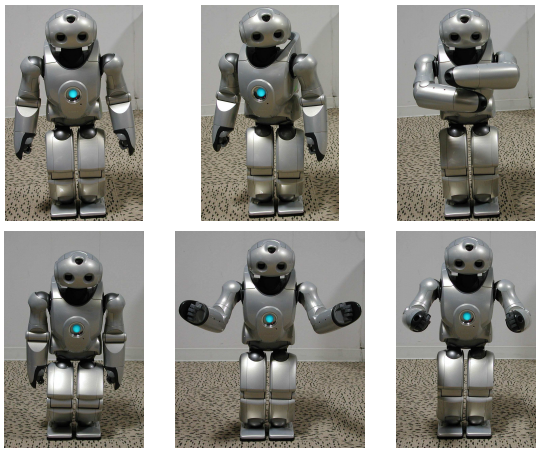


Figure 6-15: A selection of QRIO body postures generated by the overlay system. From left to right, top to bottom: normal (sociopetal) standing posture; maximum sociofugal axis; defensive arm crossing posture; submissive attentive standing posture; open, outgoing raised arm posture; defensive, pugnacious raised arm posture.

and has a positive relationship with the human, its nods and torso leans will be more pronounced; conversely, when disengaged or uninterested in the human, these illustrators will be attenuated or suppressed entirely. By adjusting the parameters of its head tracking behavior, the overlay system allows QRIO to observably alter its responsiveness to eye contact. The speed of its movements is also altered: an angry or fearful QRIO moves more rapidly, while a QRIO whose interest in the interaction is waning goes through the motions more sluggishly. All of this displayed internal information is modulated by QRIO's individual attitude, such as its extroversion or introversion. The human, already trained to interpret such signals, is able to continuously update his or her mental model of QRIO and the relationship between them.

It is clear that the latter interaction described above is richer and exhibits more variation. Of course, many of the apparent differences could equally well have been implemented through specifically designed support within the conversational interaction schema tree. However the overlay system allows these distinguishing features to be made available to existing behaviors such as the conversation schema, as well as future interactions, without such explicit design work on a repeated basis.

For the most part, the overlays generated within this implementation were able to make use of continuous-valued mapping functions from the various internal variables to the postural and proxemic outputs. For example, a continuous element such as the sociofugal/sociopetal axis is easily mapped to a derived measure of interaction desire (see Section 6.3). However, there were a number of instances in which it was necessary to select between discrete output states. This was due to the high-level motion control interface provided to us, which simplified the basic processes of generating motions on QRIO and added an additional layer of physical safety for the robot, such as prevention of falling down. On the other hand, this made some capabilities of QRIO's underlying motion architecture unavailable to us. Some expressive postures such as defensive arm crossing, for instance, did not lend themselves to continuous modulation as it could not be guaranteed via the abstraction of the high-level interface that the robot's limbs would not interfere with one another. In these cases a discrete decision function with a probabilistic component was typically used, based

on the variables used to derive the continuous postural overlays and with its output allowed to replace those overlay values directly.

In addition to discrete arm postures, it was similarly necessary to create several explicit gestures with a motion editor in order to augment QRIO's repertoire, as the high level interface was not designed for parameterization of atomic gestures. (This was presumably a safety feature, as certain parameterizations might affect the robot's stability and cause it to fall over, even when the robot is physically capable of executing the gesture over the majority of the parameter space.) The gestures created were emblematic gestures in support of proxemic activity, such as a number of beckoning gestures using various combinations of one or both arms and the fingers. Similar probabilistic discrete decision functions were used to determine when these gestures should be selected to override continuous position overlays for QRIO to adjust the proxemic state itself.

QRIO's high-level walking interface exhibited a similar trade-off in allowing easy generation of basic walking motions but correspondingly hiding access to underlying motion parameters that could otherwise have been used to produce more expressive postures and complex locomotion behaviors. Body language actions that QRIO is otherwise capable of performing, such as standing with legs akimbo, or making finely-tuned proxemic adjustments by walking along specific curved paths, were thus not available to us at this time. We therefore implemented an algorithm that used the available locomotion features to generate proxemic adjustments that appeared as natural as possible, such as uninterruptible linear walking movements to appropriate positions. The display behaviors implemented in this project should thus be viewed as a representative sample of a much larger behavior space that could be overlaid on QRIO's motor output at lower levels of motion control abstraction.

Chapter 7

Interactively Shaped Communication

By and large, humans do not communicate atomically. That is, for whatever size atom one might wish to define (e.g. a gesture, a facial expression, a transition between postures, a spoken sentence), important information is expressed in the interrelation between those atoms: the order they occur, their timing, and their relative emphasis. In Chapter 6, I discussed the automatic alteration of body language expression based on a historical context given by the robot's personality and memory. However, these expressions can still largely be thought of as atomic: although they will be interrelated by whatever continuity is present in the robot's internal drives and emotions, the parallel nature of the generating architecture renders them mostly independent. Complex communication, on the other hand, takes place in sequences, and the *way* the sequence is put together is as important as *what* is in the sequence.

We would ultimately like to make robots that communicate using the complex interplay of elements that characterizes human communication. This is currently a daunting task, due to the enormous state-space. Isolating all of the subtleties of what the interrelations between the elements "mean" into rules that the robot can be programmed with in advance is, for all intents and purposes, impossible, due to the context dependence of these meanings. Likewise, while it is somewhat attractive to consider having the robot learn to appropriately sequence its communications through some form of reinforcement learning (RL) process during its interactions with humans, this suffers the same intractability problems that are well documented in using RL to learn sequences[141]. Mainly, the credit assignment problem, compounded with the limits of human patience encountered when using RL in a large state-space with a human trainer providing the reward signal. Furthermore, in the particular case of real-time human communication, the unrepeatability of individual situations (the non-stationary environment, to use RL terminology) places further obstacles in the path of using traditional RL techniques.

I do not attempt to solve this entire problem in this thesis. Ultimately, if it is indeed solved, I predict that it will have been addressed through a combination of analysis of large amounts of human-human interaction, coupled with application of RL-inspired learning mechanisms during human-robot tutelage, in much the same way as developing humans learn and are taught to communicate as they grow up in society. Instead, I concentrate on the tutelage portion of this approach: how to teach the robot, through a face-to-face interaction, to express particular communicative sequences in the way that we wish them to be expressed. I refer to this process as "interactive shaping".

Interactive shaping addresses the refinement of two principal components of the way in which communicative elements are combined into sequences: their relative timings and styles. The immediate consequence of the process is practical: the human teacher is able to make the robot generate specific desired output without requiring recourse to off-line programming. The process has more forward-looking implications, however. It also provides a support structure for later work into meaningful expansion of the robot's social learning capabilities, such as the application of learning algorithms to attach "meaning" to the manner in which sequences are assembled, and to increase the robot's general communicative repertoire by collecting "chunks" of the learned sequences for later reuse.

This chapter therefore commences with a theoretical discussion of interactive shaping, since the term "shaping" is a term already in need of clarification in the context of human-robot interaction, and an overview of related work in the surrounding research areas. I will then discuss my approach to timing and the temporal structure of robot communication, describing my design of a timed action sequence scheduler and executer for the robot. This is followed by a similar look at style and the semantic enrichment of a robot's communicative motion capability, in which I implement a system for allowing the robot to connect human style commands to its own internal identifiers, and "shift" actions to that style when so instructed. I then present an description of an overall interactive shaping implementation, in which interactive tutelage is used to assemble and refine communicative sequences by using instruction and demonstration. Finally, I describe a novel application of the implementation: interactive specification and refinement of a "performance" of the humanoid robot Leonardo, in the same vein as a director coaching a human actor through a scene.

7.1 Shaping Human-Robot Interaction

In the specific context of communication, I use the term "interactive shaping" to describe a process of specialized meta-communication: a guided interaction in which a human teacher communicates to a robot how the robot should express a particular communication of its own. The robot learns by doing, and the teacher receives feedback on the robot's performance by observing its attempts at the communication. Like other forms of learning by progressive approximation, such as reinforcement learning or supervised learning, this is an iterative process; but it is one of compressed iteration, in which the human gives instructions as well as feedback. As such, it draws deeply from the concept of tutelage[43].

I choose to retain use of the term "shaping", however, to emphasize that the primary concern of the process is the overall *shape* of the robot's communicative output. At its heart, the process consists of rearranging a set of component parts such that the sum is different; it is a method for warping the behavioral envelope to produce a specific realization. The focus is on the communication, and its attendant fine-grained subtleties, not the learning algorithms themselves that may underpin various aspects of it. Since the teacher already has a vision of the output that is needed from the robot, it is impractical to spend time converging to that output via a method such as reinforcement learning; and due to the highly non-stationary nature of the environment and the large space of potential communications, there is no guarantee that conventional transition rules learned this way will be useful in the future (as opposed to larger fragments or "chunks" of communication, which may well be strongly suited for reuse, as long as the robot has a way to categorize them effectively).

This is not to say, of course, that machine learning techniques have no place in this process, but rather that they will have to fit in within an overall structure that places its heaviest weight on the human's instructions — for example, they might be applied towards associatively learning the relevance of the aforementioned communication chunks. This is not a major deviation from typical usages of machine learning techniques to date, which have showed the strongest performances in areas that have been carefully managed with the aid of human intervention. However, my use of the word "shaping" can be seen as significantly different from some of the senses that have previously been attached to it. I will therefore summarize those usages and place my own into the resulting context.

The term "shaping" originates from experimental psychology and the animal reinforcement learning literature. In its original sense, it refers to processes which seek to elicit specific behavior by rewarding successive approximations to the desired behavior, such as operant conditioning[271]. The first to use the term in the context of robotics was Singh, who characterized it as referring to the construction of complex behaviors from simple components[270]; in particular, the making of composite sequential decisions by concatenating basic decision tasks.

Saksida et al. later used the term to describe the process of manipulating pre-existing behaviors through carefully applied rewards in order to spawn additional new behaviors, claiming that this use of the term was more consistent with its roots in animal behaviorism[254]. They suggested that the term "chaining", also from experimental psychology, was a more closely aligned descriptor for systems like Singh's. They further identified their intentions for the concept of shaping by coining the alternative name "behavior editing", making it clear that they were referring to a process by which one behavior is converted into another. Their usage does involve interaction with a human trainer, who provides the delayed reinforcement signal, but they specifically eschew the use of explicit advice or teaching from the human; the teacher's job is only accomplished through choice of the reward criterion.

Dorigo and Colombetti, using the more specific term "robot shaping", describe a system in which immediate reinforcement is used in order to speed up the process of training efficient, simple behavior components, and then to train a module to coördinate the low-level components into complex behaviors[87]. The basic tasks the small mobile robots are trained to perform are simple hunting and avoidance behaviors, reminiscent of Braitenberg vehicles[33]. Dorigo and Colombetti do recommend exploiting the knowledge of a trainer, but due to their use of reinforcement on every time step, they discount a 'human in the loop' as too slow, recommending instead a computer program as the trainer — thus requiring the implementation of a training system that converts human knowledge into an accurate continuous evaluation mechanism to deliver appropriate and timely reinforcement. They too clarify their use of the term through the application of another descriptor, "behavior engineering", showing that their meaning of the term refers to the construction of new behaviors (in the behaviorist sense).

Perkins and Hayes also use the term "robot shaping", and are more explicit about using it to mean a combination of engineering design and evolutionary learning[230]. Their conceptual framework emphasizes human involvement in the process, in that the human should as far as possible remain free to concentrate on the high level structure and decomposition of the task, by needing to know as little as possible about the implementation details of the underlying learning components. The human's domain knowledge, transferred interactively, is important in order to increase the performance of the search-based evolutionary learning, but they do not envisage human

tutelage of the sort that is being suggested in this thesis; however they do introduce the concept of providing reinforcement as often as possible, even if not fully accurate (their term is "progress estimator"), and suggest the use of scaffolding[305] ("providing enriched learning opportunities") to help constrain the learning problem.

My term, "interactive shaping", is thus closest to the usage of Perkins and Hayes, on two accounts. First, the goal is to have the teacher concentrate on the level most appropriate to human thought, that of designing the resulting communication sequence, without having to be overly concerned with any particular learning algorithms that might lie beneath. Second, the teacher must be encouraged to give feedback (or other reinforcement or corrective signals) whenever it can be most useful, and to appropriately scaffold the learning task whenever practical. My usage of shaping is also close to the original usage of Singh, in that it concerns the assembly of complex output from simpler atomic actions. From Dorigo and Colombetti's usage I match the element of being able to shape the atomic actions themselves as well as the output sequence.

Thus of all the above examples, shaping as defined here is perhaps furthest from the original behavioral sense of the term, raising the question of whether I should have instead chosen "chaining", or something else. However I feel that this demonstrates that my usage of the term is more closely aligned with its usage in robotics, and that it now means something subtly different in robotics than it does in animal psychology; something from which I do not wish to distance this work too far. Furthermore, since I am most interested in the "shape" of the robot's physical output, rather than its underlying behavior *per se*, the term seems more appropriate than "chaining". My addition of the word "interactive" then serves to emphasize the direct involvement of the human teacher in the process, in a much more fundamental way than is shown in any of the other robot learning examples.

Why not something else, like "directing", given the similarity of what I propose to the thespian process of the same name? Unfortunately, the term has already found common usage, in the sense expressed by the phrase "goal-directed", to refer quite specifically to the process of influencing an agent's output behavior by altering its high-level goals (see Section 7.1.1), which is quite different from the process I discuss here. I have therefore chosen to stick with the term "interactive shaping", and simply ask the reader to keep in mind the contributions of other aspects drawn from the learning pantheon.

7.1.1 Other Related Work

There are of course many examples in the robotics and wider machine learning literature in which human advice is used to influence the learning process, e.g. [67, 174, 179]. However it is irrelevant to summarize this literature, because as explained above, the human interaction involved here is not intended to directly impact the learning process *per se* (for a survey of social robotics that includes social learning efforts, see [99]). Rather, the emphasis is on direct human shaping of sequential display output, and the body of robotics work in this area is somewhat slimmer.

Matarić et al. report on efforts to teach sequences through human-robot demonstration, through concatenating known primitive behaviors to match a human model[188]. Snibbe et al. present a system explicitly aimed at generating lifelike motion, in which a human designer combines and sequences behaviors using a video game controller-like interface they refer to as a "motion

compositor"[274]. This mechanism is not interactive in the sense intended here (face-to-face, natural communication). Kaplan et al. have applied "clicker training", a popular method of dog training, to the teaching of complex tasks to robots[145]. While this method is closest to the traditional behaviorist definition of the term "shaping", it therefore moves away from the added human guidance and instruction that the notion of interactiveness as expressed here is intended to introduce.

Lauria et al. introduce a technique entitled "Instruction-Based Learning" which bears some similarity to the work in this thesis in that the human instructs the robot in a task using natural language; in addition to motor-grounded phrases used directly, the human can teach the robot new phrases composed of phrases that the robot already knows[168]. However in the scenario they describe, the entire instruction occurs prior to execution of the task, so the human must mentally represent the entire task in advance. In contrast, this work involves the robot 'learning by doing', so that the human can receive immediate feedback on how the sequence is shaping up, and can make adjustments along the way. In fact, the closest related work in robotics is our own work on tutelage and collaboration[43].

Lastly, van Breemen reports the use of animation principles in designing expressive control for the Philips iCat[289]. His motivation for this work parallels the efforts in this thesis in that he acknowledges that actions based solely on algorithmic feedback loops do not produce motion that is sufficiently lifelike and communicative. He advocates instead a combination of pre-programmed motions and feedback loops. Unlike the work in this thesis, assembly of these sequences takes place off-line as in traditional animation, and a general architecture for merging the two categories on the robot has not yet been developed. The acknowledgement to the field of animation, however, is well-placed, as much of the relevant work in the area of communication shaping originates there rather than in robotics.

The animation community has had a keen interest in shaping, in the sense used here, because it collectively shares the goal of making it easier to generate highly expressive output. There has consequently been a range of efforts towards such topics as the separation of style from content (e.g. "Style Machines"[34]), and the broader area of motion parameterization (see Section 6.2 for a summary). As with the case of learning algorithms, however, the majority of the literature in this area is highly mathematically specialized and not directly relevant to the interactive aspect of this particular shaping process.

Of most interest, therefore, are those research reports that explicitly involve human interaction (in the sense used here, as opposed to human engineering) in the creation of sequenced animation elements. Earlier efforts, such as Girard's system for the generation of legged locomotive motions for animated animal characters[106], and Cohen's use of user-specified spacetime constraints[69], considered interactivity to be an end result of the motion parameterization process in that the human could then produce a desirable result by iteratively specifying parameters and constraints until the output was considered satisfactory. One avenue of progress in the further development of interactive methods for animation shaping was thus to incorporate more intuitive user interfaces (in the HCI sense) into the process, such as mapping the user's mouse and keyboard input onto the muscular actions of physically simulated characters, allowing direct interactive control of the character in real time[164].

Perhaps the closest relevant material to the style shaping work in this thesis is the work of Neff and Fiume, who describe a system for interactively editing what they term the "aesthetic" qualities of character animation[212]. They are concerned with essentially the same qualities as those that

will be discussed here: timing, and motion "shape" (which they divide into two components: "amplitude", referring to the joint ranges of a motion, and "extent", referring to how fully the character occupies space versus keeping its movement close to its body). Their conceptualization of interactivity, however, is primarily associated with how smoothly the editing tools can be embedded into the animator's workflow, rather than in allowing the animator to interact with the character as an agent to be taught on a familiar level.

Lee et al. take steps towards more natural interactive shaping of character animation by taking a corpus of motion capture data, reprocessing it for the purpose of calculating good transitions between motion sections and then allowing the human to "act" out paths through the character's resulting pose graph by performing corresponding motions live in front of a video camera[169]. Once again, however, the emphasis is on control of the character, rather than instruction of it, so that interaction really refers to the interface between the human and the computer rather than between the human and the character.

It is unsurprising that most of the work on interactive sequencing and shaping in animation treats the problem as primarily one of control efficiency, because an ordinary animated character is nothing more than a representation of the appearance of a character. More closely relevant work in terms of interactivity comes from the specialized daughter field of regular character animation: the realm of synthetic characters. Concerned with the development of lifelike virtual creatures, this field merges traditional animation with autonomy techniques from artificial life[287, 181, 182]. Once again I will concentrate here on the involvement of human interaction in synthetic character development, rather than in the development of character autonomy itself.

A number of researchers have focused on providing a facility for exerting external influence on the behavior of autonomous synthetic characters. A common motivation for doing this is to allow the behavior of autonomous characters within an interactive theatre setting to conform to progress through the plot, or to enable autonomous game characters to provide convincing responses to player interactions. Blumberg and Galyean isolate four levels at which it is desirable to allow control of character behavior to be asserted: at the motor skill level, the behavioral level, the motivational level, and the environmental level[30]. As they are concerned primarily with retaining the level of underlying autonomy of the characters, they characterize the external influences as "weighted preferences or suggestions" rather than "directorial dictates".

In general, these efforts can be divided up into the resulting level of character autonomy that results. Systems exhibiting strong autonomy, such as that of Weyhrauch, make use of sparse external influences on behavior that primarily ensure the occurrence of particular story elements[301]. Conversely, systems with weak autonomy necessitate more continuous use of fine-grained behavioral influences, such as that reported by Mateas and Stern[189]. The middle ground, semi-autonomous characters, has been pushed strongly by researchers into game-based AI such as those at the University of Michigan. Laird et al. summarize their motivation as producing intelligent characters which are human-like and engaging, with behavior driven primarily by environmental interactions and the characters' own goals[163].

The assertion of external control within these constraints is then developed by Assanie, who uses the term "directable characters" to express the goal of developing characters that can plausibly improvise behavior that complies with high-level direction from a "narrative manager"[13]. Primary considerations are thus goal coherence (on the part of the character) and the expression of a consistent personality. Magerko et al. further extend the concept of the narrative manager

into a computer-based director that manipulates the overall interactive drama from a pre-written script[183]. The scope of directorial influence is divided into "content variability", referring to *what* things occur in the story, and "temporal variability", *when* these things occur. However, these refer to plot devices, rather than the fine communicative timings and motions that are the topic of this chapter.

These projects illustrate the different meaning of the term "directing" that was mentioned earlier in the chapter. All of these mechanisms are primarily concerned with modifying the behavior of characters 'on the fly' during dramatic performance or game play, not with being able to teach the characters in advance. For the most part, the communicative skills the characters will use are already there, having been scripted in advance. It is therefore necessary to examine a different kind of interactive modification of synthetic characters — one that is in some ways more basic and historically removed from the efforts concerning the characters' intelligence — the development of interactive authoring tools for character behaviors.

Perhaps the seminal work in this area was the "Improv" system of Perlin and Goldberg at the NYU Media Research Lab, which combined a focus on expressive authoring with the end goal of real-time visual interaction with the character[231]. Their system combined an animation engine, containing procedural methods for creating realistic atomic animations and transitions between them, with a behavior engine that allowed users to script complex rules about how agents would communicate (among other things). The authors made a point of stating that with the exception of the work of Strassman, which also emphasized the need for expressive authoring (but was too early to envisage real-time visual interaction)[277], other synthetic character efforts up to this point did not seek to improve the author's ability to have the character express personality, but to improve the agent's ability to reorganize and express (within the constraints of the scenario) the personality it already has.

At around the same time, Hayes-Roth et al. were investigating along the related lines of allowing users to create compound behaviors in animated characters through sequencing and constraining basic autonomous behaviors, with the final expressive interaction being determined by the designed-in improvisation within those autonomous building blocks — a process they refer to as "directed improvisation"[125]. Their system was deployed for assessment in the form of a "Computer-Animated Improvisational Theater" testbed for children. Participants could select behaviors from a character's repertoire, concatenate them into more complicated behaviors, generalize them along certain variables, and then classify and assign descriptors to the new result. This system therefore exhibits perhaps the closest similarity with the goals of the research in this chapter. However the instructions issued by the children are not interactive in the sense of teaching, but in the sense of a traditional menu-driven graphical user interface, and the improvisation within the behaviors is free form, with no timing component.

The extension of these foundational works into the realm of using the desired ultimate real-time interaction for authoring as well as experiencing the final product, however, has been slow in coming. Taranovsky developed a system that allows the control of an animated puppet through decomposition of a task into atomic components, sequencing the components with interactive commands, and then executing the sequence on the character in order to generate a result animation[280]. This too is similar to the goals of the topic at hand, but the authoring interaction is still not a full attempt at a natural, situated interaction with the character, and the consideration of timing details is minimal.

The historical progression of synthetic character research is thus one that shows a trend away from the field's roots on the animation side, concerned with the workflow and skills of the human designer of the character's personality, and towards the foundations on the artificial life side, in terms of refining the autonomous and improvisational skills that allow the character to change and express that personality in response to internal and external conditions. That is understandable to a large degree, but it neglects a need that becomes more prominent as interactions with the character become longer term: the need to teach the character new forms of expression and their meaning. The work in this thesis chapter thus closes the loop on these prior efforts, by enabling the redirection of the desired real-time interactivity into the authoring of new interactions, through human guidance and tutelage.

7.2 Precise Timing for Subtle Communication

The timing of elements within a piece of human communication can have an enormous effect on the resulting impression made on its recipient. A socially skilled observer can infer from close analysis of timing such value judgements as fear, dishonesty or indecision (hesitation); eagerness and enthusiasm (rapid transitions between elements); deliberation or concentration (precisely periodic transitions); emphasis (temporal isolation) or dismissal (temporal crowding); and a great many more. Yet despite the power of timing, it remains a largely unexplored area in human-machine interaction. To be sure, timing is omnipresent in the literature in the form of planning and scheduling, but its importance to the purpose of expressiveness has not received much attention. Part of this is no doubt due to the immense challenge of attempting to design a machine to correctly interpret this information within communication emanating from the human; the field of machine perception is not yet ready to tackle this in a comprehensive way. However, widespread use of appropriately timed subtle communication can be seen in the field of animated entertainment, so it is an area deserving of similar attention in the field of autonomously animated characters such as interactive robots.

The bulk of the attention to precise timing in this area so far has been concerned with basic aspects of producing lifelike motion. A crucial aspect of this goal is the minimization of unnecessary delays and latencies, in order to avoid looking unnatural and stereotypically "robotic"[264, 42]. As long as responses to stimuli can be generated within an appropriate threshold, this consideration is typically seen as having been satisfied. There has also been some attention paid to the appropriate timing of actions in the context of human-machine collaboration: essentially, this means ensuring that social conventions such as turn-taking behavior appear to be shown sufficient respect by the robot[36, 38] or animated agent[58, 247]. The timing involved in observing these social norms is still coarse, however, since it is relative to the holistic action sequences of the other participant, rather than the atomic fragments of one's own compound activities.

In order to teach highly expressive communication sequences to a robot, it is therefore necessary to design a way to precisely specify not only the ordering of those sequences, but the absolute timings of the components within them. In other words, fine-grained control of the triggering times of atomic actions is needed. This is something that was not previously available in the system used to control Leonardo. In order to explain the timing additions I have made, I therefore must first explain how actions are usually triggered within the robot's behavior system.

7.2.1 Action Triggering in C5M

Our robotic behavior engine is an extension of the `C5` system developed by the Synthetic Characters group at the MIT Media Lab; we entitle this extended version `C5M`. The system is ultimately an evolutionary outgrowth of the ethology-inspired architecture for action selection originally developed by Blumberg[28]. Since then the system has been enhanced by various additions to memory, perception and learning, also largely inspired by ethology and biology. For a clear and intuitively-written high-level introduction to `C5`, see[29]. More detailed descriptions of the inner workings of the system at various stages of its development are provided in several Master's theses from the group[90, 135]. I will just summarize the components of the system relevant to action triggering.

The architecture of the behavior engine is essentially a message passing system between a collection of coöperating subsystems. The basic message unit is entitled a data record, and every piece of information must be encapsulated into such a record if it is to be transmitted elsewhere. For example, the beginning of a cycle of the behavior engine could be described as "sensing", in which stage the various sensory input modalities construct data records containing their recent detections. Immediately following this, "perception" occurs, whereby the currently passed data records are matched against a structure called the percept tree, which is a hierarchical mechanism for recognizing features in the sensory data, and generating corresponding "beliefs" about the world. This latter function occurs next, in the working memory update; beliefs are produced and matched against each other to reinforce appropriate analyses of the environmental data. Finally, the action system is updated via an action selection phase, translating the creature's overview of the world at that moment into an observable policy of interaction.

Unlike the other processing stages, action selection is not necessarily performed on every cycle of the system. This is due to the necessity for dedicated antidithering protection within the system, providing the capacity for variable-length actions in terms of time required to execute, without having to be concerned with more complex actions having correspondingly higher probabilities of being interrupted by mundane behaviors. Action selection therefore proceeds as necessary: when the action currently being performed is finishing, or when something has occurred in the world that is of sufficient significance to interrupt it.

The actual triggering of actions in this architecture revolves around the concept of action-tuples. These base action elements encode five things: (1) a trigger, which may be a percept or belief; (2) the action itself, which is a container with various activation functions and internal state information; (3) a termination condition that also may take one of several forms, such as time or a particular percept or belief; (4) an object context for the action (if the action is applicable to particular objects); and (5) an estimated reinforcement value associated with performing that action under the trigger state conditions.

The global collection of action-tuples is divided into several "action groups", ordered hierarchically by the specificity of their trigger conditions. This provides a variety of behavior categories in order to ensure compelling activity, protect against the observed behavior becoming swamped by idle conditions that might favor a certain class of action, and, conversely, allowing the preferential selection of certain types of behavior that as a result of external influences. During each action selection phase, triggered action-tuples within each group compete for activation, and the best from each group are then evaluated against one another to determine the ultimate victor.

The overall `C5M` architecture facilitates a number of different opportunities for reinforcement

learning, which I will not go into here. However, one component that is notably absent is a comprehensive mechanism for sequence learning, as Blumberg et al. acknowledge as the first missing item in their overview of the learning system[29]. This absence is primarily a result of the difficulty of the credit assignment problem in reinforcement learning. However, the application described in this chapter has three major differences from the ethological reinforcement learning applications for which the system was originally designed. First is an additional requirement: a typical sequencing mechanism that merely learns to schedule actions in the correct order will not suffice. Instead, there is also the additional need for precise timing — the sequencer must be able to schedule actions at the exact moments that they are desired. Second is another requirement: the sequencer must support iterative refinement. Both the order and timing of component actions must be adjustable during subsequent shaping interactions with the human teacher. The third difference, however, fortunately makes the task easier: since the robot will have the benefit of human tutelage, the sequence can be *taught*, rather than inferred from a delayed reward. This greatly simplifies the credit assignment problem in this application.

I have therefore proposed and implemented the *Scheduled Action Group* as an additional category of action group for the `C5M` architecture. Actions can be freely added to these groups during interaction with the robot, such as during teaching. Moreover, actions that are added to a scheduled action group have relationships to other actions already in the group that can be specified, altered, or inferred from the context in which they were added. The Scheduled Action Group is itself an action, and can be triggered via the action selection process like any other action in the robot's repertoire. When a Scheduled Action Group is triggered, it decomposes the relationships between its component actions into a definite schedule, and begins triggering those component actions at the appropriate times according to that schedule for as long as the action group remains active. Precisely how this scheduling is implemented is described in the following subsection.

7.2.2 Temporal Representation and Manipulation

The most basic design decision in developing a mechanism for the robot to schedule atomic actions into precisely timed compound actions is what should be the underlying representation of time. This decision does not refer to the choice of units of elapsed time, such as whether to use world-grounded units of seconds and milliseconds, or machine-grounded cycles of the computational architecture (called "ticks" in our robot's behavior engine). Rather, the decision concerns how to represent the available knowledge about the characteristic timings of the component actions, such as potential start points and durations, in a way that best permits reasoning about the relations of these timings to one another — whatever the fundamental units.

Since reasoning about time is an important factor in addressing many computational problems, there have been many different methods for representing knowledge about time (see [5] for an introduction). I will start this discussion by setting out the particular requirements that a method must satisfy in order to be suitable for this application, and then give a brief overview of common methods and why they may or may not be appropriate, before giving a detailed presentation of the method that I ultimately selected. First, however, I will define some terms. A *point* or an *event* is an instantaneous moment in time, similar to a point on the real number line in that it has a position but no extent. An *interval* is the contiguous stretch of time from one point to a different point; the earlier of the two points is the interval's *start point*, the later of the two is the interval's *end point*, and the resulting extent or distance between them is the *duration* of the interval. For

the purposes of systems implemented on digital computers — such as our robot — in which as a result the fundamental measurement of time is discretized, these definitions assume that points may only be placed precisely at the locations of discrete time samples.

The purpose of incorporating temporal knowledge into our robot is to support its capability to learn precisely timed communicative sequences from an iterative process of human instruction. A method of representing that knowledge should therefore ideally possess the following desirable characteristics (partially adapted from [3]):

- The ability to be constructed using uncertain information. When a new action is added to the sequence, the robot may not know exactly when it should be performed relative to all other actions in the sequence, or the human may not care. Ultimately, the robot will have to decide when to perform each action, but this need not be specified at the outset.
- The ability to represent imprecise knowledge. Although the content of an action may be known, the precise timing details of its employment under any particular conditions may not immediately be available to the robot. For example, the duration of an action may be dependent on the style in which it is undertaken (e.g. aggressively versus lethargically).
- The ability to inexpensively propagate alterations to the existing content of the knowledge base. This partially follows from the previous two requirements: as uncertain information becomes more certain, and estimates of imprecise quantities are adjusted, these changes may affect the arrangement and timing of other elements within the sequence. In addition, the iterative nature of the human teaching process means that even formerly certain information may change, and elements in the sequence may be reordered.
- The ability to permit specification of temporal relationships other than ordering. Examples of these are concurrence and mutual exclusion: in the former case, that a set of actions must take place together (whether or not their start or end points are aligned), and in the latter, that a set of actions must not be performed together (regardless of the relative order in which they do in fact occur). Since it is subject to the physical constraints of the world, the need for a robot to be able to enforce such types of coördination on its actions is clear.

Perhaps the most obvious temporal representation, since it is what humans use in scheduling activities on a calendar, is simply to assign values to the start points and durations (or end points) of events. This form of representation is known as "Dating", and makes it computationally easy to calculate ordering information. However this application largely requires the reverse computation: given the ordering specified by the teacher, find appropriate trigger times for the actions. Moreover, it is not well suited to the representation of uncertainty and imprecise knowledge. Examples of the use of the technique in artificial intelligence can nevertheless be found, e.g. [127] and [142].

Alternate representations have been proposed that are directed more towards temporal logic. The "Situation Calculus" of McCarthy and Hayes represented temporal knowledge within a collection of world states that described what was true at each instant in time[191]. Shoham later developed a point-based logic formalism[267]. Such systems make it difficult to distinguish between the effects of simultaneous events.

State-space representations have been used to incorporate temporal information in a more compressed fashion, e.g. [97, 252]. States represent the condition of the world at particular time

instants, and actions the transitions between these states. The duration of an action is the duration of its corresponding state transition, and the time spent in a particular state is the duration of the state. These representations are not well suited to the representation of uncertainty in action order, however, nor concurrence and mutual exclusion.

The nature of the problem domain — representing the mutual temporal relationships between collections of events — may suggest the potential advantages of network representations, and indeed a number have been proposed. Some of these focus on the representation of duration, such as PERT networks[184]. Dean and McDermott proposed a method of representing relations between events, in terms of limiting bounds on the possible duration between them[78]. Ordering can be preserved in the presence of inexact information if bounds are allowed to extend to infinity ($\pm\infty$). Reasoning over these networks is dependent on graph search techniques, which can become combinatorically prohibitive as the number of events increases.

Another class of network representation that has been widely used to encode temporal knowledge is constraint networks[201]. In general, constraint networks are networks in which the nodes are variables, unary constraints on the nodes restrict the domains of these variables, and the edges connecting the nodes specify binary constraints between the variables. To actually find a specific set of values for the variables (a *solution*), or the exact set of possible values (the *minimal domain*), the network must be solved for the constraints. Various collections of algorithms for solving and working with constraint networks are available[208, 160, 79].

The earliest approaches in applying these networks to temporal representation used the networks' nodes to represent events, and the binary constraints to specify ordering relationships — an arrangement which eventually came to be called "point algebras"[185, 291, 292]. Representing temporal relationships in this way suffers from a number of problems, as set out by Allen[3]. Briefly, the main issues are difficulties in representing intervals, such as whether they are open or closed at their endpoints, and the inability to represent mutual exclusion with binary ordering constraints.

Allen therefore proposed a competing constraint network representation based on using time intervals as the network nodes rather than events — this arrangement is entitled "interval algebra"[3, 4]. Representing intervals directly avoids endpoint uncertainty[7], and the temporal constraint primitives available for intervals are much more expressive than those for events (see Section 7.2.2.1). These constraints make it easy to represent concurrence and mutual exclusion, and weak constraints can be constructed from disjunctions of primitive constraints in order to represent incomplete information about the relationships between intervals.

The constraint network representation itself has the further advantage that the precision of individual pieces of information is not critical until the network needs to be solved. Actually solving constraint networks, however, can be computationally problematic, and constraint propagation algorithms for interval algebras are typically exponential in the number of constraints. However, when conditions exist such that heuristics can be used to alleviate this, the architecture can lend itself to powerful mechanisms for internal rearrangement and change propagation.

The application implemented in this chapter has a strong need to represent uncertain and imprecise information, as well as temporal constraints beyond simple ordering, both of which are significantly addressed by the interval algebra representation. The application also has aspects that can alleviate much of the computational expense of a constraint network architecture, as will be explained shortly. An interval algebra network (IA-network) representation was therefore selected as

the internal representation of the Scheduled Action Group. More theoretical and implementational details of this network representation are given in the following section.

7.2.2.1 Interval Algebra Networks for Precise Action Timing

The basic units of the interval algebra representation are time intervals. Although time intervals do ultimately have start and end points, the time values of these do not need to be considered in order to perform reasoning, as they do in point algebras. Instead, the temporal relationships between time intervals are defined in terms of primitive relationships that collectively express all possible qualitative positions of the intervals' start and end points relative to one another. There are 13 of these primitive relations. They are *before* (b), *meet* (m), *start* (s), *finish* (f), *overlap* (o), and *during* (d); the inverses *after* (or *i-before*, ib), *met-by* (or *i-meet*, im), *started-by* (or *i-start*, is), *finished-by* (or *i-finish*, if), *overlapped-by* (or *i-overlap*, io), and *contains* (or *i-during*, id); and *equal* (e). These primitive relations are visualized in the graphical representation of Figure 7-1 (adapted from [3]).

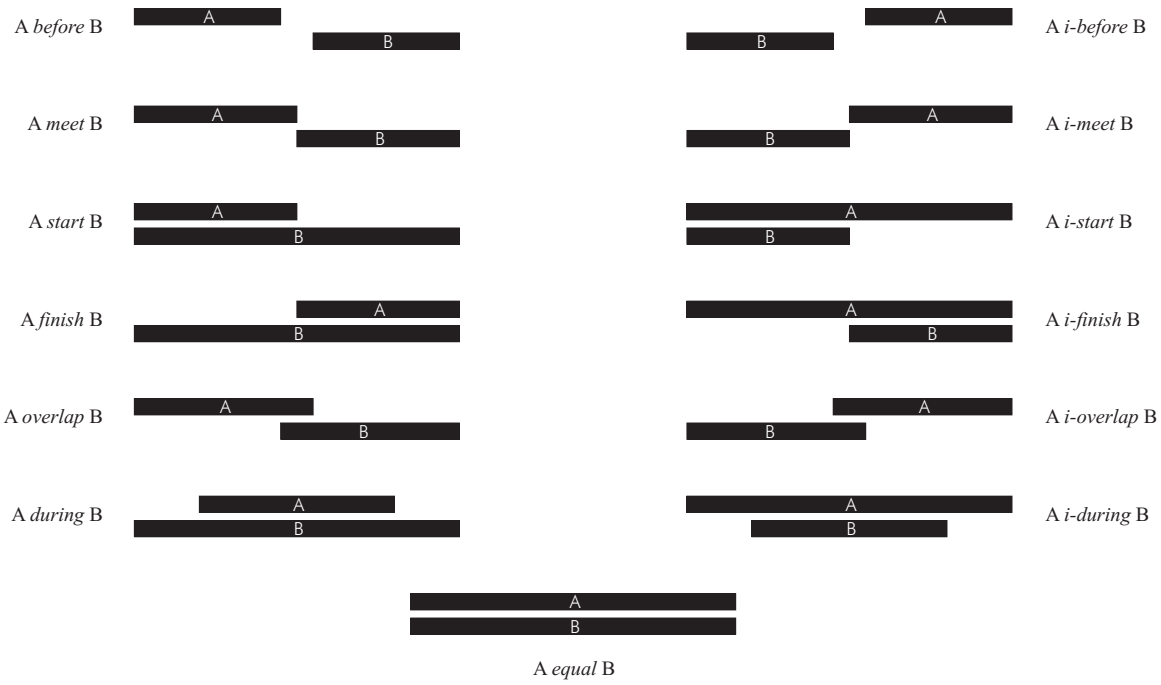


Figure 7-1: The thirteen primitive interval relations.

The interval algebra representation is a constraint network in which the nodes are variables which each occupy a time interval (e.g. a single action). They are often referred to as interval algebra networks (IA-networks) or Allen interval networks. The set of possible intervals in real time for a node is its domain, or the unary constraints on the node. The edges of the network are binary temporal constraints between pairs of nodes. These binary constraints consist of disjunctions (unions) of the primitive temporal relationships. For example, if it is known that two intervals should start at the same time, but the durations of the intervals are not known and thus the time order of the end points is uncertain (or unimportant), the relationship constraint from one to the

other is given by $\{s, is, e\}$. For a small example network representing a compound robot action, see Figure 7-2.

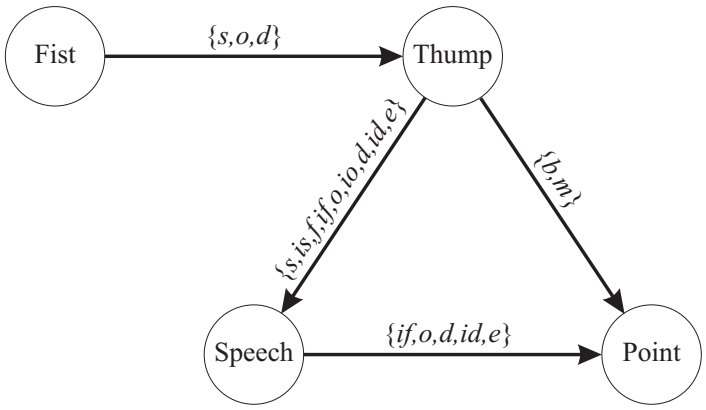


Figure 7-2: An example interval algebra network representing the possible timings of a compound robot action.

This example represents a scenario in which a robot orator delivers a portion of a fiery speech. In this fragment, the robot thumps its fist on the podium in order to accentuate the piece being spoken, and then further emphasizes its words by pointing its fist out towards the audience. The interval network describes the potential relations between balling the hand into a fist, moving the arm to thump it on the podium, moving the mouth to speak the words, and finally moving the arm forward to point the fist outwards. According to the network, the making of the fist must occur concurrently with the start of the thumping movement (so that the fist is complete before the hand hits the podium); the thump must occur concurrently with the speech (the relationships *before*, *after*, *meet* and *met-by* are excluded); the speech must occur somewhat concurrently with the point motion; and the point motion must not commence until the thump motion has concluded.

Although this example network is depicted as a directed graph, this is generally a matter of illustration convention. The direction of the edges merely explains the dependency ordering of the printed set of relationships: they are relationships "from" the source node "to" the destination node. Reverse edges, which contain the inverse set of relationships, are implicit and typically not shown. In the example in the paragraph above, the reverse edge between two intervals constrained to start at the same time contains the same set of primitive relations, because it contains the inverse pair $\{s, is\}$ and the equals relation, e, which is its own inverse. Similarly, unconstrained relationships are shown with no edge at all for clarity of illustration; in reality, this relationship implies an arc containing a disjunction of all the primitives. Basic interval networks are thus essentially undirected, and can be drawn with equal correctness in the reverse orientation (Figure 7-3).

Let us denote by $\mathbf{A}$ the set of all primitive relations, and by $\mathcal{A}$ the set of all possible binary constraints between intervals, represented as disjunctions of elements of $\mathbf{A}$. The number of elements of $\mathcal{A}$ is thus:

$$\|\mathcal{A}\| = 2^{\|\mathbf{A}\|} - 1 = 2^{13} - 1 = 8191$$

Note that the empty set, $\{\emptyset\}$, has no meaning in the context of temporal constraints (the

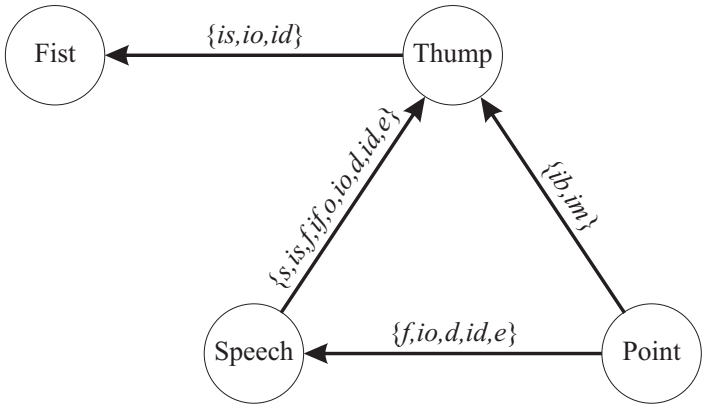


Figure 7-3: The example interval algebra network of Figure 7-2 illustrated according to its inherent inverse relations.

tightest possible temporal constraint is a single element set containing one primitive relation from $\mathbf{A}$), so it has been explicitly subtracted from the calculation of $\|\mathcal{A}\|$ above.

Consider the situation in which time intervals ι are represented as closed continuous segments of the integer number line $\mathbb{Z}$ that have non-zero length.[1] Let $\mathbf{X}$ be an IA-network consisting of n nodes $x_1, \ldots, x_n$. Each node x_i has an associated unary constraint $\mathbf{u}_i$ which contains the (non-empty) set of possible time intervals that may be assigned to x_i; the set of all such constraints in $\mathbf{X}$ is labeled $\mathbf{U}$. Similarly, each pair of nodes $(x_i, x_j : i < j)$ has an associated binary constraint $\mathbf{b}_{ij}$ which is an element of $\mathcal{A}$; the set of these constraints in $\mathbf{X}$ is labeled $\mathbf{B}$ (the reverse constraints, $\mathbf{b}_{ij}$, are implicit). A *solution* to $\mathbf{X}$ is an assignment of a specific time interval ι_i to each x_i such that it satisfies the associated unary constraint $\mathbf{u}_i$ and that collectively all of them satisfy $\mathbf{B}$. The network $\mathbf{X}$ is said to be *consistent* if at least one such solution exists for the given $\mathbf{U}$ and $\mathbf{B}$.

For the application of scheduling precisely timed action sequences on the robot, it is insufficient simply to determine whether or not the network is consistent; an actual solution must be found (which also proves the network consistent in the process). It is most straightforward to find a solution if the constraints on the network are as restrictive as possible given the available information. The *minimal domain* of $\mathbf{X}$ is defined as the situation when every set $\mathbf{u}_i$ in $\mathbf{U}$ is reduced to its smallest possible coverage of $\mathbb{Z}$ ($\sum_{-\infty}^{\infty} \iota \in \mathbf{u}_i$ is minimized; alternatively stated, $\mathbf{u}_i$ represents the minimal domain of x_i) such that $\mathbf{X}$ remains consistent under $\mathbf{B}$.

A number of algorithms exists for computing the minimal domain of an IA-network — see [195] for an overview of methods that can be used in the presence of qualitative and quantitative constraints. Unfortunately, computation of the minimal domain of a general IA-network is NP-hard in the number of constraints[291]. This can make them impractical for use in long-term scheduling problems. While this is true for the general case, however, special case restrictions on the structure and size of individual networks, and approximation algorithms that relax the kinds of solutions that are required, can be used to avoid the worst case computational scenarios[288].

Pinhanez and Bobick, for example, developed a system for constraining action recognition based on the temporal plausibility that particular actions might be currently occurring[238]. Since this

[1]Other representations are possible; this was chosen based on [6] in order to best represent the discrete time samples of a digital computer.

did not require finding an exact solution, but merely determining whether or not a solution could exist that placed a given action within an interval of interest, comparisons could be generalized to the special intervals "past", "now" and "future". This formalism was given the corresponding name "PNF Calculus"[237]. Unfortunately, the generation of precise robot communication does require a specific solution.

A common approach to dealing with the combinatorial explosion as IA-network sizes get larger is simply to divide the network into independent clusters with limited interconnection among them. This technique was employed by Malik and Binford in order to ensure adequate performance of their worst-case-exponential "Simplex" algorithm[185], and an automatic clustering algorithm (based on Allen's reference interval strategy[3]) was developed by Koomen[155].

Fortunately, bidirectional social communication is an application that features inherent temporal clustering, due to the presence of turn-taking behavior[253, 104]. The number of atomic actions that must be independently scheduled is partially bounded by the length of the communicative expression that contains them, and this in turn is bounded by the frequency of arrival of instances in which it is the human's "turn" to communicate. Since individual Scheduled Action Groups never become larger than the number of actions between these consecutive human inputs, the probability that the system will encounter a computational hazard is significantly reduced. However, in anticipation of the potential for long, involved communication sequences, other optimizations are also exploited.

The first such optimization is the strengthening of the binary constraints between nodes in the network, since this has the effect of reducing the search space when computing the minimal domain. When a new action is added to a Scheduled Action Group, the binary constraints between its interval and those of the pre-existing actions are, by necessity, initially weak. This is because the constraints are initially set to represent the uncertainty in the temporal relationship by only excluding those relations that are certain not to apply. For example, a new action that is defined as starting at the same time as a pre-existing action is given the binary constraint $\{s, is, e\}$ since the durations of the two actions, and thus the relative end point positions of their intervals, are not known. Similarly, the temporal constraints between the new interval and the other pre-existing intervals in the network are not yet known, so the relationship is set to the set of all possible primitive constraints, $\mathbf{A}$.

These constraints can be strengthened by removing temporal relations that are superfluous. Relations are superfluous if their removal would not result in the loss of any solutions of the network. For example, the network shown in Figure 7-2 has at least one superfluous relation. Since the thump motion must conclude at or before the commencement of the point motion, yet the relations *before* and *meet* are not present in the relationship between the thump motion and the speech, it clearly follows that the speech interval and the point interval can not be equal. Thus, the primitive *equal* can be removed from the set of relations between the speech and the point motion, since that will not cause any loss of a consistent solution.

Removal of all superfluous relations is known as *closure*, and can be accomplished in IA-networks via a process of constraint propagation. However, closure has itself been shown to be an NP-complete problem[292]. Rather than a full closure, therefore, an approximate closure is performed using Allen's *transitive closure* algorithm[3], a special case of the path consistency algorithm for constraint satisfaction[201, 178] that operates in polynomial time. The path consistency algorithm has the property of losslessness: although an approximation, it will not result in the removal of any

valid solutions to the network. Rather, it will simply not remove all of the superfluous relations. However, it removes enough of them to be worthwhile.

This algorithm works by examining sub-networks of three nodes, and reducing the binary constraints between each these nodes to the minimal sets required to maintain the consistency of the sub-network. This can be accomplished with the aid of the transitive closure of pairs of individual primitive relations $(r, q) \in \mathbf{A}^2$. Consider three intervals: A, B and C, arranged such that $A\ r\ B$ and $B\ q\ C$. The transitive closure of r and q is the set of all possible relations ρ that could exist been A and C in this situation: $A\ \rho\ C : \rho \in tc(r, q) \subseteq \mathcal{A}$. Since there are only $\|\mathbf{A}\| \cdot \|\mathbf{A}\| = 169$ possibilities for the transitive closures of all primitive relations, these can easily be stored in a lookup table (Table 7.1, from [3]).

r \ q	b	ib	m	im	s	is	f	if	o	io	d	id	e
b	b	*[all]*	b	$b\ m$ $s\ o\ d$	b	b	$b\ m$ $s\ o\ d$	b	b	$b\ m$ $s\ o\ d$	$b\ m$ $s\ o\ d$	b	b
ib	*[all]*	ib	$ib\ im$ $f\ io\ d$	ib	$ib\ im$ $f\ io\ d$	ib	ib	ib	$ib\ im$ $f\ io\ d$	ib	$ib\ im$ $f\ io\ d$	ib	ib
m	b	$ib\ im$ $is\ io\ id$	b	$f\ if\ e$	m	m	$s\ o\ d$	b	b	$s\ o\ d$	$s\ o\ d$	b	m
im	$b\ m$ $if\ o\ id$	ib	$s\ is\ e$	ib	$f\ io\ d$	ib	im	im	$f\ io\ d$	ib	$f\ io\ d$	ib	im
s	b	ib	b	im	s	$s\ is\ e$	d	$b\ m\ o$	$b\ m\ o$	$f\ io\ d$	d	$b\ m$ $if\ o\ id$	s
is	$b\ m$ $if\ o\ id$	ib	$if\ o\ id$	im	$s\ is\ e$	is	io	id	$if\ o\ id$	io	$f\ io\ d$	id	is
f	b	ib	m	ib	d	$ib\ im\ io$	f	$f\ if\ e$	$s\ o\ d$	$ib\ im\ io$	d	$ib\ im$ $is\ io\ id$	f
if	b	$ib\ im$ $is\ io\ id$	m	$is\ io\ id$	o	id	$f\ if\ e$	if	o	$is\ io\ id$	$s\ o\ d$	id	if
o	b	$ib\ im$ $is\ io\ id$	b	$is\ io\ id$	o	$if\ o\ id$	$s\ o\ d$	$b\ m\ o$	$b\ m\ o$	$s\ is\ f$ $if\ o\ io$ $d\ id\ e$	$s\ o\ d$	$b\ m\ if$ $o\ id$	o
io	$b\ m\ if$ $o\ id$	ib	$if\ o\ id$	ib	$f\ io\ d$	$ib\ im\ io$	io	$is\ io\ id$	$s\ is\ f$ $if\ o\ io$ $d\ id\ e$	$ib\ im\ io$	$f\ io\ d$	$ib\ im$ $is\ io\ id$	io
d	b	ib	b	ib	d	$ib\ im$ $f\ io\ d$	d	$b\ m$ $s\ o\ d$	$b\ m$ $s\ o\ d$	$ib\ im$ $f\ io\ d$	d	*[all]*	d
id	$b\ m\ if$ $o\ id$	$ib\ im$ $is\ io\ id$	$if\ o\ id$	$is\ io\ id$	$if\ o\ id$	id	$is\ io\ id$	id	$if\ o\ id$	$is\ io\ id$	$s\ is\ f$ $if\ o\ io$ $d\ id\ e$	id	id
e	b	ib	m	im	s	is	f	if	o	io	d	id	e

Table 7.1: Lookup table for the transitive closure $tc(r, q)$ between the primitive interval relations.

Furthermore, it is easy to use the transitive closures of primitive relations to compute the transitive closure of two sets of primitive primitive relations $(\mathbf{R}, \mathbf{Q}) \in \mathcal{A}^2$. The transitive closure in this case, $TC(\mathbf{R}, \mathbf{Q})$, is just the union of the transitive closures of all possible pairings of the primitive relationships in $\mathbf{R}$ and $\mathbf{Q}$:

$$TC(\mathbf{R}, \mathbf{Q}) = \bigcup_{\substack{r \in \mathbf{R} \\ q \in \mathbf{Q}}} tc(r, q)$$

The complete constraint propagation algorithm for achieving 3-node path consistency in the network using the transitive closure can be found in [292]. Briefly, each time a change in the binary constraint between two nodes, $\mathbf{b}_{ij}$, is asserted, this pair is added to a queue. When it becomes time to find the closure of the network, the node pairs are removed from the queue in order. For each pair (i, j) so removed, the algorithm determines whether the current value of the current binary constraint $\mathbf{b}_{ij}$ can be used to further constrain the binary constraints $\mathbf{b}_{ik}$ and $\mathbf{b}_{kj}$ for any other node k in the network. If so, the constraint is adjusted and the relevant pair of nodes added to the tail of the queue. The algorithm completes when the queue is exhausted.

Since this algorithm produces path consistency only on subnetworks of 3 nodes, it is not able to remove some superfluous relations that result from constraints across subnetworks of 4 or more nodes. Vilain et al developed an algorithm for 4-node path consistency that also executes in (higher order) polynomial time[292], but it was not considered necessary for this application due to the presence of other optimizing factors. The results of applying this algorithm to the example network from Figure 7-2 are shown in Figure 7-4.

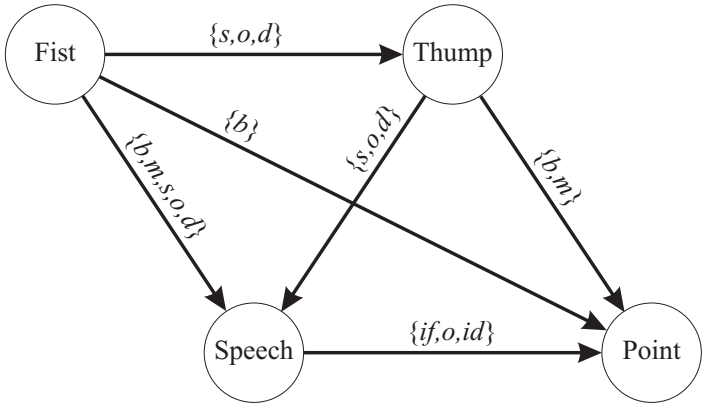


Figure 7-4: The example interval algebra network of Figure 7-2 having been closed using Allen's path consistency algorithm. Note that several superfluous relations have been removed, and that the network is now fully connected.

The second optimization that is performed concerns the unary constraints on the node intervals. In this real time interactive application, absolute time values have no special meaning, and thus there is no need to constrain intervals to any particular region of $\mathbb{Z}$ — it is only their positions relative to the start point of the sequence that matters. As a result, no unary constraints are placed on the nodes prior to starting computation of a solution to the network. First the approximate closure of the network is computed, as described above, and then a variant of the reference interval

method is used to begin computing the minimal domain of the network[3].

At this point, we are able to make use of the fact that the intervals in question represent the execution of actions in the real world. Their duration can therefore be estimated — and measured. With upper and lower bound estimates of the duration of an interval, it is possible to place upper and lower bounds on the start and end points of the intervals themselves, if anything at all can be known about them from the temporal constraint relations. Intervals such as these — whose start and end points are bounded, but in between these values no point can be excluded as a potential start or end point — are said to exhibit continuous endpoint uncertainty, and networks composed of these intervals have been shown to be a more tractable special case[292].

This can be understood by considering the source of a great deal of the combinatorics in IA-network representations: the fragmentation of the unary constraints into sets of disjoint time segments caused by mutual exclusion. For example, if the binary constraint between two intervals A and B is given by the set $\{b, ib\}$ — i.e., that A must not 'touch' B — the unary constraint on A is now split into two disjoint segments that must be considered separately. If this can be avoided, by only applying lower bounds to interval start points and upper bounds to interval end points, continuous endpoint uncertainty can be maintained. This can be the case when duration information is known, because this information can be used to compute more accurate bounds — for example, time segments that might otherwise need to be considered part of the unary constraint can be pruned if they are not sufficiently long to contain the interval in question.

In order to support this optimization, it is necessary to maintain information about the duration of atomic actions. This, of course, can vary based on the style and context in which the action is actually executed, which the robot does not necessarily know in advance (see Section 7.3). However, interactive shaping is an iterative process, and the robot "learns by doing" — it tries out actions as it adds them to the sequence that it is learning. The robot therefore generates a rough idea of the duration of the action at this time, and this information is stored within an addition to the standard action encapsulation (Section 7.2.1) called a *self-timed action*.

This initial timing evaluation does not account for later variation in the action's duration as it is shaped, however, so when it comes time to refine the timing schedule the robot "self-simulates" or "imagines" the sequence playing out whenever this occurs (Section 3.3). Variations in the duration of the action are incorporated into the timing model of that specific action in this specific sequence via the cloning mechanism that ensures the unique identity of atomic actions within Scheduled Action Groups. See Section 7.4.3.2 for a description of the support for this process.

The final optimization that operates in the favor of this application is simply that there is no need to compute the set of possible solutions to the IA-network in order to produce the desired output timing; only one solution is required, provided that it is generated based on a consistent rule that ensures that the robot's behavior is consistent from the point of view of the human. The rule that was selected was as follows. Once the approximation to the minimal domain has been computed, intervals will be assigned such that their start points occur at the earliest feasible opportunity, unless there is no lower bound on the start point, in which case they will be assigned such that their start points occur as close as possible to the earliest bounded start point in the network. In any case, if actions are temporally underconstrained and thus show up in inappropriate places, the human can simply add additional constraints during the interactive shaping process (Section 7.4).

Although the basic interval network architecture confers a number of advantages that have

already been elaborated upon, it is still primarily a qualitative representation. For example, constraining two intervals by the the temporal primitive A *after* B does not specify anything about how long the 'gap' interval should be between the end point of B and the start point of A. Since a repeatable rule for the determination of a solution from the minimal domain, as described above, it is completely predictable when A will in fact start: it will start precisely one time unit after the end point of B. Under the basic IA-network model that has been implemented, there is no way to force A to start any later, other than synchronizing it to an existing interval that itself ends later than B. In order to represent the precise timings between actions that we require for communication, an extension needed to be added to the basic model.

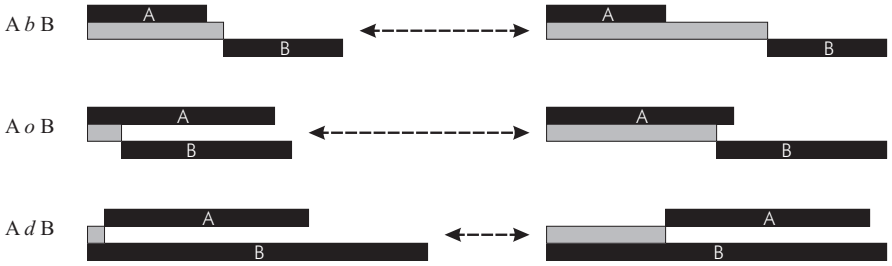


Figure 7-5: Primitive interval relations for which precise relative start point timing is not inherently specified. The same is true of the inverses of each of these illustrated primitives.

Not all of the primitive relations require any extension in order to deliver precise timing, however. Relations which imply synchronized endpoints already exhibit single time unit precision: if two intervals are constrained by the *start* relation, they will certainly commence at precisely the same time point. The primitives that require the extension are those in which there is an implied variable length interval between the start points of two intervals constrained by them, that can be adjusted without altering the constraint (Figure 7-5). Therefore, this is exactly how I have extended the basic model to represent precise interstitial timing: by providing an additional subsystem for inserting and managing variable length "spacer" intervals within the network (Figure 7-6).

Figure 7-6: An example of the use of a spacer interval to specify precise relative start point timing. The relation between intervals A and B does not change, but the duration of the spacer interval S can then be used to precisely control the earliest relative start of B within the constraints of the original relationship.

Inspired by the use of "rubber" space in the typesetting software LaTeX, the duration of spacer intervals can be altered as desired during the shaping process; they are also transparently added and removed as necessary, and automatically resized when interval relations change in certain ways. In order to prevent the proliferation of fragmented or orphaned spacer intervals, and to keep the implementation details hidden from the human user, the use of spacer intervals follows a set of simple rules.

- Each action interval may have at most one associated spacer interval.
- The constraint relation between a spacer interval and its action interval is always $\{m\}$. The spacer interval can therefore also be thought of as a "delay" interval.

- Spacer intervals are represented by regular action intervals whose payloads are null actions. Since intervals cannot have zero length, if an interval with an associated spacer interval has its start point synchronized to another interval's end point the spacer interval is deleted.
- A spacer interval between two action intervals cannot be allowed to violate the temporal constraints between those two intervals. If this occurs, both the constraints and the spacer interval are modified automatically such that the length of the spacer interval is minimized. The absence of constraint violations is verified through mental self-simulation.

The following figures illustrate the use of spacer intervals to provide precise control of timing. Spacer intervals are initially inserted when the interval constraints do not adequately represent the desired output timing (Figure 7-7). If a spacer interval is used to move an action interval across a temporal constraint boundary, the spacer interval's duration is truncated and the constraint relation updated to reflect the same output result (Figure 7-8). If the constraints between action intervals are altered such that spacing is no longer necessary, the spacer interval is removed (Figure 7-9).

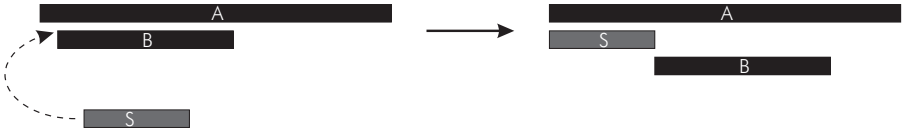


Figure 7-7: Insertion of a spacer interval S in order to precisely control the start point of B relative to A.

Figure 7-8: Automatic alteration of the duration and relations of spacer interval S to achieve minimal duration, performed when an adjustment to the duration of S changes the primitive interval relations between A and B.

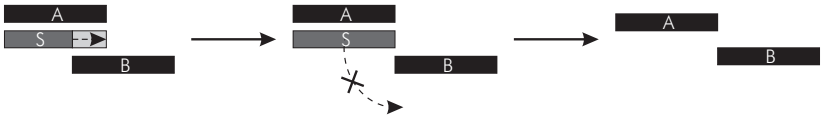


Figure 7-9: Automatic deletion of spacer interval S when implicit endpoint synchronization of intervals A and B would otherwise cause it to have zero duration.

Since the relationship of the spacer interval and the associated action interval is so tightly constrained, and since the same temporal reasoning functions are used to process spacer intervals as action intervals, the computational overhead associated with spacer intervals is negligible. How the system copes with explicit manipulation of the lengths of spacer intervals is addressed in the following section.

7.2.2.2 Support for Interactive Shaping of Time

The way in which information is entered into this interactive robotic system renders it substantially dissimilar to many conventional problems in constraint network reasoning. It is true that the

abstract process overview is similar: data and constraints are entered, a solution is produced, and then the data and constraints may be modified to produce a related but different (and hopefully, improved) solution. However, the manner in which this modification takes place — interactive shaping — makes a profound difference. We want the interaction to be natural, and for the human teacher not to have to know too much about the underlying architecture of intervals, durations and relations. Therefore, when changes are made to the network, they must take place within a local context that is understood by both the human and the robot, so that values to be altered are correctly altered and values to be preserved are preserved.

To illustrate this point, I will examine how timing changes are made to the extended IA-network that underlies the Scheduled Action Group, from the simplest modifications to the more complex. Primarily, this requires intervals to be able to 'move around' in time. This activity will demonstrate that an additional extension to the architecture is needed in order to allow timing to be interactively shaped in a genuinely natural fashion.

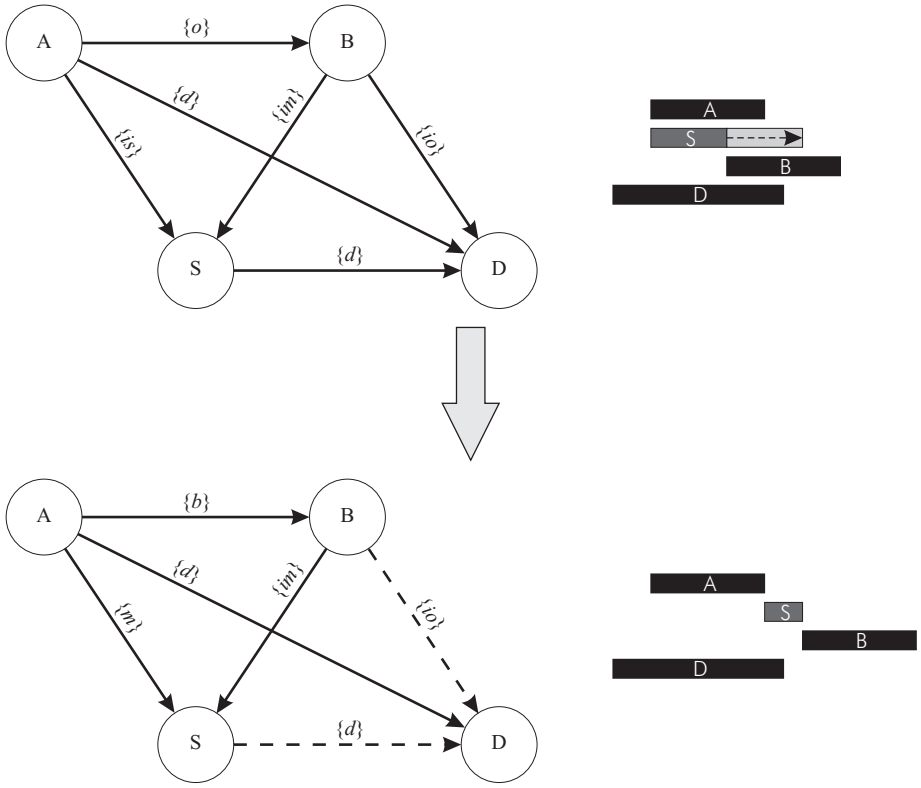


Figure 7-10: An example of network inconsistency due to local alterations. Movement of interval B with respect to interval A can invalidate relations to intervals elsewhere in the network, such as D.

Adjustment of the relative timing of intervals within the bounds of their existing interval constraints has already been seen, in Section 7.2.2.1. This is simply a matter of making the appropriate numeric change to the duration of the spacer interval. However when relative timing changes force changes in the temporal constraints between two intervals, things become more complicated, especially once the binary constraints in the network have been strengthened through the closure process. Since the temporal relationship between any two nodes have been validated in

terms of the relations between those nodes and their neighbors, a change in one of those neighboring relations can invalidate the consistency of the network, as shown in Figure 7-10. Changes in temporal constraints, therefore, must be propagated throughout the network.

When I assert that a single interval A "moves" relative to another interval B, the intuitive meaning of this statement is that the primitive relation describing their relative temporal positions changes: $A\ r\ B \rightarrow A\ r'\ B : r, r' \in \mathbf{A}; r \neq r'$. This, of course, is only certain when the temporal relation between A and B is known. Before an IA-network is solved, on the other hand, the exact relation between the two intervals that will be produced in the solution can be uncertain. It is therefore more appropriate to think of moving as introducing the potential for a primitive relation between A and B to exist in some new solution to the network that could not have existed in any prior solution. More formally, if A is represented by the network node x_i and B is represented by x_j, A moves relative to B if $\mathbf{b}_{ij} \rightarrow \mathbf{b}'_{ij}$ such that $\exists\ r : \mathbf{b}'_{ij} \ni r \notin \mathbf{b}_{ij}$.

The moving of an interval in this sense is dealt with in almost exactly the same way as the path consistency algorithm for strengthening the binary constraints of the network: by constraint propagation using the transitive closure mechanism across 3-node subnetworks. Consider any third node x_k that forms a 3-node subnetwork with the nodes in question above. Since x_j represents the "reference interval" relative to which the interval represented by x_i is moving, its relationship $\mathbf{b}_{jk}$ to x_k can for the moment be considered static. Similarly, the new relationship $\mathbf{b}_{ij}$ is already known. Therefore a lossless approximation to the new relationship $\mathbf{b}'_{ik}$ is simply given by the transitive closure function $\mathbf{b}'_{ik} = TC(\mathbf{b}_{ij}, \mathbf{b}_{jk})$.

Thought of in this way, as a single interval moving within the network, reveals the potential for an additional optimization that takes advantage of the unidirectional flow of time. Depending on the exact nature of the change in relations expressing the original movement, a significant portion of the network may be able to be excluded from the constraint propagation calculations without risking inconsistency in the network.

This is easy to see by considering a change in which the interval A moves strictly *later* relative to interval B. For example, if A was originally constrained to start while B is active and finish before B does, the temporal relation between A and B would have been $\{s, d\}$ (Figure 7-11). If this constraint was then altered such that A must instead start *after* B and *finish* while B is active, the temporal relation becomes $\{im, f, io, d\}$ (Figure 7-12). It is easy to see that this change fulfils the definition of moving, and although there remains some overlap in terms of potential placement of A, it certainly cannot move to a position that is any earlier (relative to B) than it could have occupied before. Therefore, any interval that was already constrained to end before the start point of A — that is, any interval whose relation to A is $\{b\}$ — cannot possibly have been affected by the movement of A, and can be safely excluded from consistency calculations in this case (Figure 7-13).

Similarly, in some cases in which an interval moves earlier, other intervals can be excluded from the consistency check depending on their own relationship to the "reference" interval. The complete set of rules for partitioning the network under interval movement is given in Table 7.2. This optimization may seem trivial for the movement of single intervals, but if large numbers of intervals are to be moved simultaneously, it can result in worthwhile savings.[2] How and why multiple intervals might be moved simultaneously foreshadows the main point of this section.

[2]The observation that time flow produces a graph ordering was similarly used in the "Sequence Graph" technique of Jürgen Dorn to produce a large reduction in complexity[88].

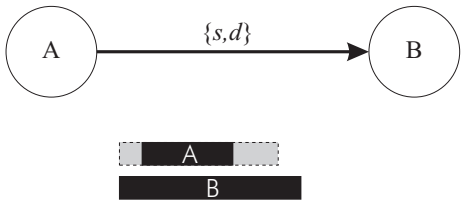


Figure 7-11: An example of the possible domain of an interval A subject to the given set of primitive relations to interval B, requiring that A start while B is active and finish before B finishes.

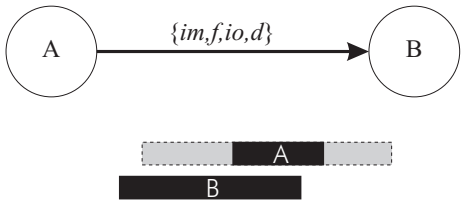


Figure 7-12: The resulting possible domain of interval A (from Figure 7-11) if its relationship to B is changed such that it must now begin after interval B, and finish while B is active.

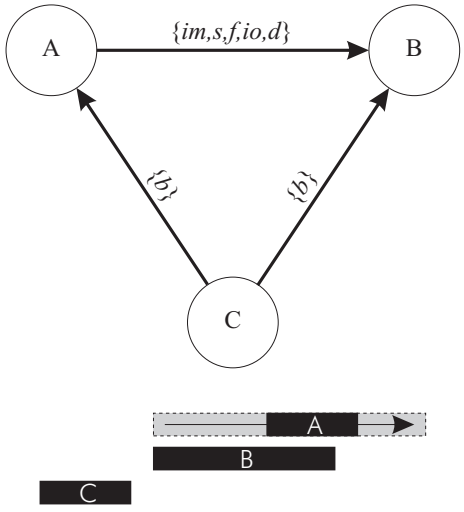


Figure 7-13: An example of an interval C in the network such that the alteration to the possible domain of interval A relative to interval B (Figures 7-11 and 7-12) is guaranteed to affect neither its relationship to A nor to B.

Child interval moves earlier		**Child interval moves later**	
Other relation to parent	Unaffected by these new child-parent relations	Other relation to parent	Unaffected by these new child-parent relations
b	*im, s, is, f, io, d, e*	*ib*	*m, s, f, if, o, d, e*
m	*im, f, io, d*	*im*	*m, s, o, d*
s	*im*	*f*	*m*
o	*im*	*io*	*m*
d	*im*	*d*	*m*

Table 7.2: Relations of other intervals to a parent interval that can be used to computationally prune the closure of the network under the specified movements of child intervals of that parent.

All of this discussion is predicated on the answer to one question: why would interval A above be considered to be moving relative to interval B, and not vice versa? To put it another way, the temporal relations are bidirectional, so how is a reference interval chosen? This question can be reframed by examining the local context of the interactive shaping process that was referred to at the start of this section. During shaping, the human is trying to teach the robot to improve the timing of a sequence it already knows. If moving one element improves the timing relative to an earlier element, but simultaneously messes up the timing relative to a later element, the human is no better off with this process than just modeling the entire sequence ahead of time and specifying the precise absolute timings — and probably worse off, since it will likely be a frustrating journey to this conclusion. When an interval is moved, therefore, it is not enough for the system just to maintain internal consistency. It needs to know which other intervals should also move; i.e., which temporal relationships should be preserved, and which should be updated. The network needs a hierarchy.

There is not a great deal of research relating to this specific aspect of IA-network design. Since these representations are primarily used in scheduling systems and AI applications for reasoning about particular temporal relations in order to find *optimal* or *possible* solutions, perhaps this application represents a novel problem domain: the manipulation of IA-networks to produce *specific* solutions, where the optimization criterion is not directly related to some aspect of the solution itself but is instead concerned with minimization of the external manipulations required.

That said, directed constraint networks and relative ordering constraints have certainly been a feature of research for some time, typically for reasoning about causality and duration (e.g. [80, 10, 9]). A typical application is scheduling in the presence of precedence constraints, for example determining whether a process composed of a set of tasks whose order is partially dependent on certain resources can complete within a specified length of time.

Extensions in which the intervals themselves have direction have also been proposed for use in applications of IA-networks to non-temporal problems. For example, consider a spatial problem concerning the positioning of vehicles on a two-way road, with the intervals representing the vehicles which thus have both a spatial extent and a direction of travel[246].

More recently, Ragni and Scivos have proposed a "Dependency Calculus", applicable to constraint networks in general including IA-networks, for reasoning about complex dependencies between events, causal relations, and consequences[243]. However, once again, the focus is on reason-

ing to extract knowledge from the network, rather than on manipulating the network to produce a specific solution.

Extending the IA-network architecture for the Scheduled Action Group to incorporate hierarchical information was conceptually simple. The network was actually implemented as a directed graph rather than merely being depicted as such. The direction of an edge between two nodes thus conveys the "control" precedence, and the content of the edge continues to convey the set of temporal constraint relations. The hierarchy is defined such that the source of the edge is considered the parent and the destination the child. In addition to hierarchical information, this provides a compact and intuitive representation for disambiguating the sense in which the temporal constraints apply: from the parent to the child. To get the relations from the child to the parent, the set can be inexpensively inverted.

While this representation is simple, the key is to be able to apply it in a meaningful way — in other words, to best represent whatever actual hierarchy is present in the action sequence. Once again, this is done by taking advantage of the structure inherent in the human tutelage process. When the human teaches the robot a new action using a reference to an existing one, the system uses this referential dependence as a clue to a contextual dependence. Thus it sets the referent to be the hierarchical parent, and the new action to be the child.

The human does not always refer to actions in the explicit context of other actions, however. In this case, the system takes advantage of the temporal context of the teaching process in order to make an estimate of the appropriate parent action. As the robot is "learning by doing", good choices for parent actions are likely to be those that have been recently learnt, or ones that are nearby (and causally preceding) the site of temporal attention in the sequence during shaping. The latter, of course, requires that the robot has already computed a solution to the Scheduled Action Group network at least once. This has indeed occurred, via the robot's mental simulation procedure. For more details on the interactive shaping process, see Section 7.4.

The final detail to consider is how the procedure for making temporal constraint adjustments — i.e., moving intervals — is modified to accommodate the hierarchical information. Again, this is conceptually very straightforward. Essentially, when a parent node is moved, its temporal constraint relations with its children are preserved; when a child node is moved, its constraint relations with its parent are updated. A simple example is shown in Figure 7-14. The constraint propagation function is able to manage all of the details without major alterations to its algorithm, because it is already making a decision whether or not to apply constraint propagation to particular nodes based on whether or not the time dependence of the move can potentially affect them, as described above.

One aspect of this rule that should be mentioned explicitly is the treatment of spacer intervals. When a spacer interval is added to an action it is added as a child, and spacer intervals already in the network can never become parent intervals — they are excluded from all heuristic-based selection processes. Thus the temporal constraint relation of a spacer interval to its action can never be altered — it will always move with its action in a sensible way. This shows an advantage of using these special separate intervals to represent delays, rather than a more simplistic scheme such as adding the delay directly to the beginning of an action itself: other actions can still be synchronized to the actual start of an action, regardless of the presence or length of its spacer interval. Similarly, since spacer intervals are always synchronized to the start point of another interval (or the "zero" start point of the Scheduled Action Group in the special case of a deliberate

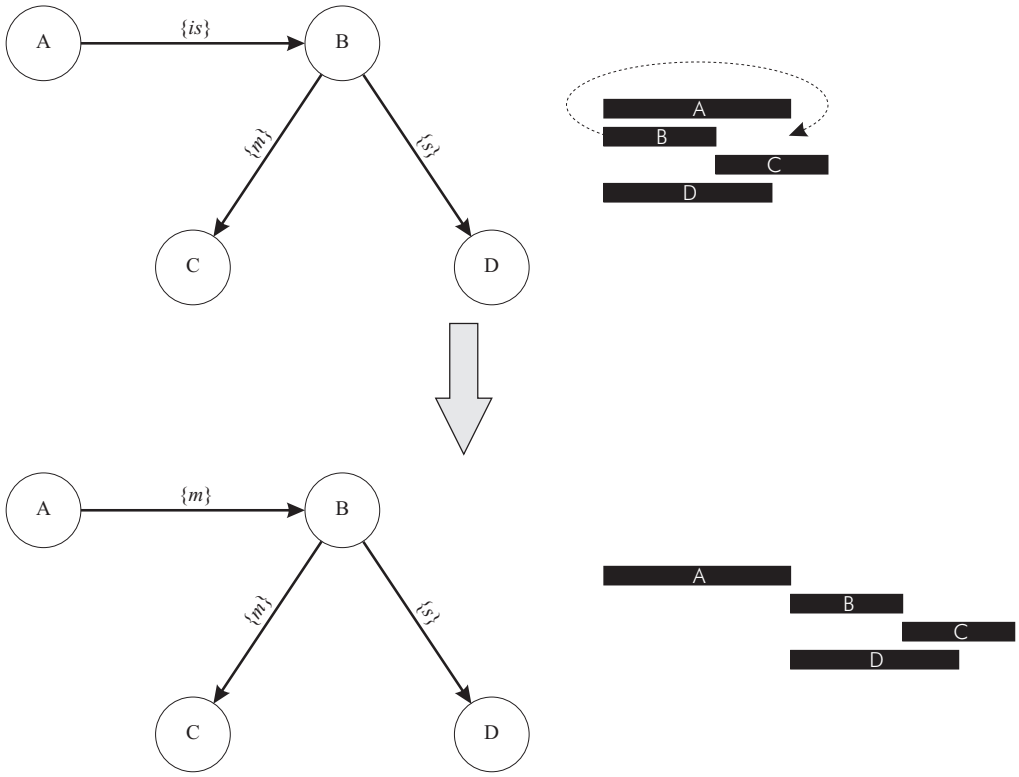


Figure 7-14: An example of interval movement in a hierarchical IA-network. When interval B is moved relative to interval A, the children of interval B are moved along with it, preserving their relationships to the parent interval.

initial hesitation), it is never necessary to refer to a spacer interval for synchronization purposes. An illustration of temporal shaping in the presence of spacer intervals is shown in Figure 7-15.

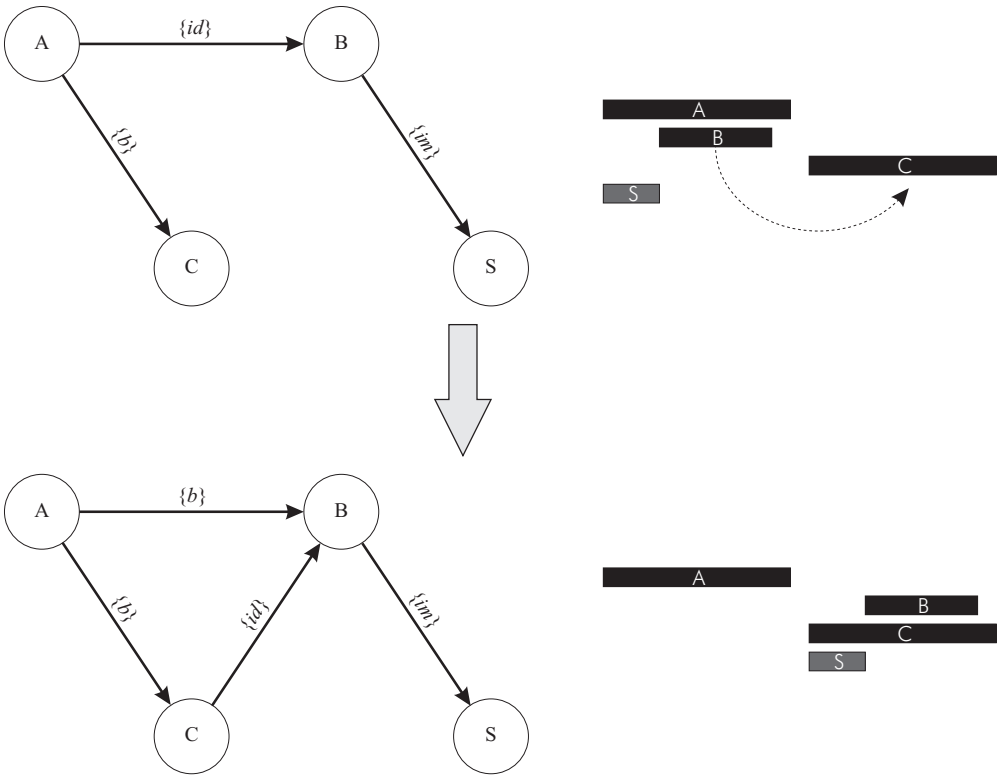


Figure 7-15: An example of interval movement in a hierarchical IA-network with the presence of spacer intervals. When interval B is moved to an explicit interval relationship with a new interval C in which its spacer interval remains meaningful, its spacer interval is similarly moved according to its child hierarchical relationship.

7.3 Fine Tuning Expressive Style

There are many subtle elements that can affect the meaning of a communication sequence. Emotions can be deliberately exaggerated or stifled to convey context beyond the face value of the emotions themselves; body movements can be spatially expansive or constrained (Section 6.1.2, and [212]); gestures can be performed intensely or blandly, and so on. Timing, in the sense of the temporal arrangement of atomic components, has been addressed. Many of the remaining elements can be thought of collectively as "style". Treating them in such a broad conceptual sense makes classification easy, as long as there is a practical way to also deal with them in an implementation.

To reiterate, the goal of the application is to allow a human to teach the robot a communicative sequence through an interactive shaping process. In order to be able to support a collective approach to style in this process, the treatment of style must be able to address most if not all of the kinds of style that might need to be represented. Humans have so many kinds of styles, though, and differing concepts of what a particular style physically entails based on cultural norms — one

person's 'forcefulness' is another person's 'arrogance', for example — that this might seem hopeless. If we can't even agree on how to realize one style, how can we aspire to realize all of them? In an abstract, disconnected learning process, we could not. However, in the interactive shaping case, we have a human 'in the loop' to ground the robot's usage of style.

In order to actually do this, the design of the system must address three core questions. First, how does the human communicate stylistic information, which of course is represented in the human's mind in human terms, to a robot which may or may not have an existing representation of that information that matches the human's? Second, how does the human specify the particular part of the complex communication sequence that should be affected? Third, how does the robot then represent that stylistic information and apply it to its internal motion model of the aspect in question? This section details the work towards addressing these three issues.

These questions provoke some obvious suggestions, which will be briefly dealt with before proceeding. To the first question, clearly a potential solution is imitation — the human can ground the stylistic meaning by demonstrating it to the robot. Imitation is dealt with elsewhere in this thesis (see Sections 3.1, 7.4.1 and 7.4.3.1), but problems with the approach include ensuring that the robot understands which aspect of the demonstration represents the desired improvement (consequently just shifting the stylistic representation issue to a lower level), and dealing with an imperfect human demonstrator (who may not even be able to do the action being taught in the first place, if the robot's morphology or performance differs too drastically). Imitation can be a powerful component of a solution to this issue, but there needs to be another mechanism to back it up.

In the context of the chapter so far, the general form to the answer to the second question is obvious: the desired communication is composed of atomic actions, so the question is already simplified into a question of how to draw the robot's attention to the correct one (or few). Having rephrased the question into one of attention rather than partition, it is also obvious that we can extract some benefit from the robot's internal attention system that has already been described (Section 3.4). Nevertheless, there will need to be additional support structure in place for occasions in which the robot's immediate introspection cannot disambiguate the action referent.

To the third question, it has already been made clear that one way is to use parametric approaches to mathematically alter underlying motion primitives (see Sections 6.2 and 7.1.1). However, these approaches are often difficult, and sometimes simply prove impossible to ensure valid results, because the underlying dimensionality of the space is so large and therefore cannot be exhaustively verified. More importantly, they must be designed in advance, or learned so intensively that to do so within the interactive shaping process would represent an infeasible diversion. Once again, parametric approaches have their uses, but additional tools are needed in the kit.

At this point I must explicitly draw a distinction between the material in this section and the "behavioral overlay" techniques of Chapter 6. At first glance, they may seem to be addressing the same goal: modifying the robot's existing behavior to express information encoded by a particular style. The difference is in the way the goal is manifested. The behavioral overlay technique allows run-time expression of information about the robot's instantaneously computed state (taking into account the memory and personality elements, of course), but it does not have any permanent affect on the underlying behavior itself. It is a mechanism for adding a communications channel. Conversely, the encoding of style information into an action via the shaping process represents a plan for expressing this information independent of the robot's state. It is a mechanism for refining an existing communications channel.

Of course, these mechanisms are not mutually exclusive — behavioral overlays can make further run-time modifications to a shaped sequence, and overlays can be permanently attached to individual actions as an means of achieving style shaping. In fact, this latter method is precisely one of the components that have been used in the implementation described in Section 7.3.2.2.

7.3.1 Motion Representation in C5M

Following on from the discussion of action triggering in Section 7.2.1, I will present a brief summary of the way in which underlying motion information is represented in the robot's behavior system (`C5M`). Although there is a tendency to use the words interchangeably, in `C5M` an *action* is not the same as a *motion*; the former refers to a conceptual resolution to "do" something, whereas the latter means a physical expression of a sequence of positions of the robot's joint motors. When triggered, actions can cause motions by essentially requesting that a particular motion takes place. However the motor sequences themselves are represented completely separately from the action system, and are handled by a different subsystem called the motor system.

In fact, the robot typically has several different motor systems, representing different areas of the robot's body, at the discretion of the behavior designer. This arrangement grew out of a crude approach to precisely one of the problems this chapter addresses: how to allow an interplay of separate stored motion sequences on one set of physical actuators, without forcing each of those stored sequences to completely represent the motion of the complete set of actuators. This implementation detail is not important to the discussion in this chapter, but it is ultimately the mechanism that allows resource contention on Leonardo to be handled such that the work presented here is possible.

Stored motions are represented by sequences of joint angles over subsets of the robot's joints; in other words, animations of parts of the robot's skeleton model, and in fact the motions are typically created offline by animating a graphical model of the robot. This representation has two immediate consequences: stored motions cannot control joints that they were not originally designed to control, and they have an implied length, represented by the number of joint positions stored in the sequence.

As a result of the former aspect, motions that are sequenced together by the motor system (as opposed to the shaping system described in this thesis, which has no such restriction) must have been designed for identical sets of joints, to prevent undefined transitions that could have physically unpredictable (and therefore dangerous) expression on the robot. This means that general forms of style shaping cannot take place at the motion level. They must instead take place at the action level.

The latter aspect means that the speed of a particular movement is fixed at the motion level, not the action level, because in order to adjust its speed a motion must be resampled. Typically, motions are sampled to a generally acceptable speed when the robot's motor system is first set up, so that various actions which may trigger the motion will generate a consistent output. While there exists support within the motor system for resampling motions at any time, doing so results in a persistent change to the motion — any action that subsequently requests that motion will then produce it at the updated speed.

These two facets of the motor system needed to be dealt with in order to implement the overall style shaping process. The ways in which they were handled is described in sections further on.

It is impossible to generate premanufactured animations for every possible motion the robot might have to perform, and of course this would not be desirable even if it was possible. The robot's motor systems are therefore designed to perform automatic blending between motions. There are two main classes of activity in which motion blending is performed. The first is in generating transitions between motions, because the robot's pose at the end of one motion is unlikely to exactly match its pose at the start of another. The second class is in generating novel motions by interpolating between several other motions, in order to allow the robot to compactly represent certain general motor skills such as reaching for an object.

When a motor system for the robot is constructed, specific motions are assigned to it, and the allowable transitions between those motions are defined. This has the effect of connecting the motions into a "verb graph" (from [136], after the verb-adverb motion parametric technique of Rose et al.[250]), which the motor system can then traverse in order to produce the output motion. If the verb graph is fully connected, the robot can produce any sequence of motions, by transitioning directly from one to the next. If it is not fully connected, for example if it is desired to restrict certain transitions, the motor system will find the shortest path from the current motion to the next desired motion, which may require playing through one of the other motions in order to get there.

Conversely, to interpolate between multiple motions, the motions are set up as the elements in a logical array that has an associated array of rational-valued blend weights that sum to unity. Again, this is done during the setup of the motor system. The resulting motor output that occurs upon the initiation of this motion array is therefore a linear combination of the input motions weighted by the blend values. Again inspired by Rose et al., a particular combination of blend weights is called an "adverb".

As is often a caveat associated with motion parameterization, it is likely that not all possible linear combinations of a given set of input motions will produce reasonable-looking motor output. In addition, the relationship between blend weight combinations that produce good output and the real world conditions in which they should be computed may not be intuitive. The motor system therefore incorporates mapping functions entitled "adverb converters", which essentially convert n-dimensional world information into m-dimensional blend weights, for some values of n and m. For example, to convert the real world spatial coördinates of an object into a blend between set of directed reaching motions conceptually arranged into a 3x3 grid, $n = 3$ and $m = 9$.

The above overview summarizes the basic state of motion representation in `C5M` prior to the extensions I implemented to support interactive shaping. I will only detail one of these in this section; the others will be covered in the following sections. This particular extension concerns direct feedback loop control of the robot's joints. From the above summary, it is apparent that motion control of the robot is highly abstracted: there is usually no facility for behaviors to directly control motors. For the most part this is eminently reasonable, because directly modifying a joint that is simultaneously part of a blended set of stored motions could cause unpredictable results.

However, there are situations in which precise control of joints is more efficient than setting up blends. One example is control of the robot's fingers. Leonardo has 4 fingers on a hand, and thus there are $2^4 = 16$ possible extreme hand poses based on whether each finger is fully extended or fully curled. Therefore, in order to produce movements between arbitrary finger positions, 32 corresponding source motions would be needed (one set for each hand). This is no longer a compact or intuitive representation.

I therefore implemented a *direct joint motor system* for C5M, which allows actions to interface directly with a particular motor system to control a subset of the joints to which that motor system has resource privileges. When activated, the system takes note of the robot's current pose, and attempts to make a smooth (linearly filtered) transition from it to the desired direct joint pose. Similarly, when an action that uses the direct joint motor system terminates, it monitors the pose which the regular motor system intends to assume, and smoothly transitions back. This arrangement assures that fine, precise joint movements can be made when necessary.

7.3.2 Representation and Manipulation of Style

From the description above of the way motion itself is represented, the starting point for style representation is fairly clear. Most of the motion atoms, with the exception of a few elements of direct control, are not generated by the robot — they have been generated in advance by a human designer. Thus the style of an atomic motion within the robot's pool of available motions is best thought of as some qualitative aspect of the difference between that motion and other existing motions in the pool that could be said to be part of the same class.

The judgement of whether actions are in the same class or not can be subjective, as can the judgement of precisely what qualitative aspect might represent a particular style. Consider, for example, a hand motion that a human might interpret to be a wave. What aspect of a hand motion defines a wave? It is difficult to narrow down; some waves have periodic motion, and some do not; some waves involve finger motion, and some do not; etc. Similarly, a human might describe one waving motion as having a more intense style than another. What aspect of the motion makes it more intense? Potential candidates might be the speed; the amplitude; the acceleration profile; etc. The subjective aspect of style that led to its treatment via a collective approach carries down to the level of stored motions as well.

This is not to say that the human animator who is designing the motions in the first place does not have a sense of the various classes and stylistic ranges that are present in the robot's motion pool. Of course, the pool of motions will be assembled with careful consideration of the types and potential variations of movements that the robot is likely to need. Furthermore, the behavior designer is already considering the modification of motions along what could be termed linear *style axes* when constructing motion adverb blend sets. The missing component here is a mechanism of mapping motions to similar style axes when they are not suitable for automatic linear composition. Therefore, there needs to be a way for the designer to group discrete motions into classes, and then to implement the style axes by connecting them into hierarchical arrangements of particular styles.

This linear style axis model is the fundamental abstract representation of style within the action architecture. From the point of view of the human, it is designed to be seamless across motion representation. That is, when an action is refined to have a different style — i.e., the motion that it produces when triggered is updated to a different motion that better expresses whatever qualities embody said style — it should not be necessary for the human to care whether the change is generated by altering the weights of a motion adverb blend, or selecting a different motion from a discrete set, or adjusting the parameters of a direct joint control system.

Before discussing how this is implemented, this abstraction raises a more immediate question. Though abstraction has clear advantages in terms of the level of architectural knowledge required by the person interacting with the robot, it also produces a potential disconnection between the

semantic understanding of stylistic concepts on the part of the robot designer versus that of the robot user. The interface to that disconnection is, of course, the robot itself. Put more concretely, how do the human teacher and the robot agree on the terminology to use during the learning interaction, if it cannot be assured that the teacher agrees with the designer? This is the problem of symbolically grounding the style representation, and it will be examined first.

7.3.2.1 Grounding Style with Interactively Learned Symbols

The general problem of how an agent can ensure that the symbols in a symbolic system are appropriately connected with their corresponding semantic interpretations is well known in AI, and is referred to as the *Symbol Grounding Problem* (SGP)[124]. Searle's Chinese Room Argument is often used to illustrate the SGP — the essence of the argument is the improbability of a non-Chinese speaker successfully learning the Chinese language by consulting a Chinese-Chinese dictionary[266]. For good overviews of the work that has been done addressing various aspects of the SGP, see [315] and [279].

The SGP is important in robotics, because the ability to physically ground symbols in aspects of the real world — as interfaced with by the robot in terms of its sensors and actuators — has been seen as a potential route to a solution. This is complicated, however, by a central aspect of the full SGP known as the *zero semantical commitment condition*, or Z condition. The Z condition essentially forbids innatism (the pre-existence of semantic resources) and externalism (the transfer of semantic resources from another semantically proficient entity) from being part of any solution to the SGP. Since this component of the thesis specifically concerns the enabling of externalism via a human teacher (and additionally makes heavy use of innatism), I am therefore not interested in tackling the full SGP.

Nevertheless, I bring it up because some relevant related work in robotics does claim to address the SGP, though it also violates the Z condition as pointed out in [279]. Billard and Dautenhahn propose an approach to the SGP based on imitation learning and socially situated communication[23, 24]. The task of the robots is to learn a proto-language through interaction with a teacher; the teacher has a complete knowledge of the vocabulary from the start, whereas the robots have no initial knowledge of the vocabulary. The grounding of the teacher's symbols takes place by association with the robot's sensor-actuator state space. This differs from the work presented here in that the teacher here is only assumed to have *a* vocabulary, not *the* vocabulary — another teacher may use a different one, and the robot should be able to cope — and the robot is not completely without an initial vocabulary — it has the vocabulary of the motion designer, which must be mapped to that of the teacher.

In other related work, Bailey et al. developed a system that can be taught to distinguish stylistically different verbs (e.g. "push" versus "shove") via action control structures called "x-schemas"[17]. X-schemas are networks of action primitives, akin to Scheduled Action Groups. In addition, high level control parameters can specify overall attributes of x-schemas. A verb is defined by its x-schema and control parameters, with the control parameters representing the stylistic component (or "adverb"). The work in this thesis differs primarily in granularity: adverbal symbols are applied to primitives, rather than overall schemas, enabling communicative sequences to be shaped with the use of multiple styles.

Roy et al. also developed a framework for grounding words in terms of structured networks

of primitives experienced by a robotic manipulator, "Ripley"[251]. The primitives used are a combination of both motor and sensor primitives. In contrast, I do not attempt to ground style multimodally. My goal in terms of grounding is not to represent symbols experientially, but simply to use experience ("learning by doing") to assist in simultaneously grounding pairs of symbols (the teacher's and the designer's) in logical axes that hierarchically connect motor expressions.

How does the human explain to the robot what constitutes a particular style during "learning by doing"? The general form of this process involves the human coaching the robot into doing something, and then applying feedback to correct any mistakes that the robot makes. If the robot doesn't know anything about what the human is asking it to do, choosing something to try amounts to making a guess. I have therefore designed a system that allows the robot to make an "educated" guess about what the human might want, based on what the robot has been asked to do in the past.

This is accomplished with a framework in which the robot makes a series of semantic hypotheses about what the human's symbols (the "labels") might correspond to in terms of the robot's internal style axis representations (the "adverbs"). When the human uses an unfamiliar label, the robot selects an adverb that it considers to be the most likely match based on an examination of the results of earlier hypotheses. The connection of this new label with the selected adverb is then added to the database of hypotheses, with a support level of zero. When the robot performs the action, feedback from the human is used to adjust the support level of the hypothesis up or down. Thus the database of hypotheses represents the prior probabilities of particular label-adverb correspondences, and the robot's beliefs of what labels correspond to which adverbs is continually adjusted to incorporate new evidence according to the Bayesian approach.

The general algorithm for hypothesizing an adverb A for a given label L is as follows:

if $\exists$ prior hypotheses for L:
    find $A = \arg\max$ (Hypothesis(L, A))
    if Hypothesis$(L, A) > 0$:
        use A
    else ($\nexists$ good prior hypotheses):
        if $\exists$ A s.t. $\nexists$ Hypothesis(L, A):
            select $A = \arg\min$ (Hypothesis$(\hat{L}, A)$), $\forall \hat{L} \neq L$
        else (all adverbs have hypotheses for L):
            use A
else ($\nexists$ prior hypotheses for L):
    if $\exists$ A s.t. $\nexists$ Hypothesis(L, A) $\forall L$:
        use A
    else ($\nexists$ unhypothesized A):
        select $A = \arg\min$ (Hypothesis$(\hat{L}, A)$), $\forall \hat{L}$

The way that this hypothesis system deals with positive reinforcement is typical: if the robot has experienced positive results from executing a particular adverb in response to a given label in the past, the likelihood that it will continue to respond to that label in the same way is increased. In addition, though, since the robot already has a finite set of style axes, it is also able to make use of exclusion learning. If many of its adverbs already have positively reinforced correspondences

with labels, a new label that the robot has not previously experienced is considered more likely to correspond to an adverb that has not been considered for labeling at all. Thus the robot initially attempts to maximize the hypothesis coverage of its adverb space, because hypothesizing over the adverb space is equivalent to communicating the extent of the space to the human (Figure 7-16).

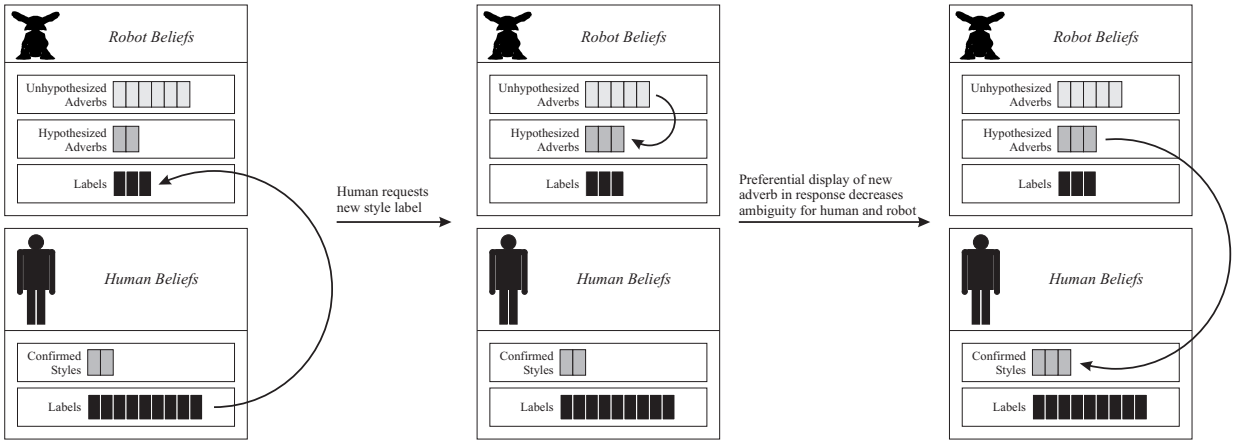


Figure 7-16: Maximizing the hypothesis coverage of the robot's adverb space serves a communicative function because each hypothesis generation is accompanied by a demonstrative trial. As well as showing the robot's repertoire to the human, this sets the stage for targeted disambiguation such as corrective feedback.

When the coverage of the adverb space is complete, the robot attempts to reason about new labels using the degree of support that existing hypotheses have received. Hypotheses having a large amount of support can be indicative not just of the correctness of the hypothesis, but of the human's predisposition to use that label to refer to a particular style. Hypotheses that are weakly supported can be examples of adverbs that have been correctly matched with that label, but also share semantic correspondence with multiple symbols used by the human. The system therefore supports many-to-one labeling of adverbs, and the ability for the human to change the preferred labeling of an adverb, by preferentially selecting adverbs with weakly supported hypotheses to assign to new label hypotheses.

Clearly the robot must express an adverb-modified action at least once in response to each label that the human uses. In the best case scenario, if the robot was somehow able to guess *a priori* the correct style axis direction for each label the human used, for an interaction using l labels, the robot would only need to express l test actions as each one would immediately be correct. Conversely, if the robot randomly selected adverbs for each label, the worst case scenario would involve the robot trying each of the a adverbs in its repertoire for every new label, resulting in al test actions needing to be expressed before the robot has a complete label mapping. Using the heuristics above, the best case remains the robot getting it right first time, but the worst case (assuming the human does not make any mistakes) is significantly reduced, $al - \frac{1}{2}(l^2 - l)$, by Gauss' formula for arithmetic series.

However, since a human teacher is present, the situation can be further improved upon. Each time a new label is applied, the robot generates and tests a hypothesis, and receives feedback from the human teacher as to the validity of that hypothesis. But the teacher does not need to be limited to giving feedback on the hypothesis; the robot has also performed a physical action, so in the presence of errors the teacher can directly ground the action that was performed, as part of the feedback that is given. In other words, if the robot does the wrong thing, in addition to inhibiting

that specific error from occurring again, the teacher can simultaneously cause the robot to generate a new, known correct hypothesis, as in the following example:

Teacher: "Robot, do that action more flamboyantly."

Robot: *[attempts the action in a different style]*

Teacher: "No, that's more *clumsy*."

Each negative hypothesis thus results in an additional positive hypothesis for a label that may be used in the future. The best case scenario in this case remains the same as before, but the worst case scenario has been altered both qualitatively and quantitatively. The qualitative change is that the worst case scenario is no longer the one in which the robot tries every adverb in its repertoire before correctly selecting the one corresponding to the new label. If this occurs now, and corrective feedback is applied each time, the robot will have achieved a complete one-to-one labeling of labels to adverbs in a test trials. If $l > a$, the many-to-one worst case (assuming the teacher ceases providing corrective feedback after the first a labels) is $a(l - a + 1)$, which is clearly better than al for all non-trivial cases ($a > 1$).

This situation is similar to an alternative strategy, which would be to reverse the interaction in the first place, and have the robot babble through its style axes initially and have the human label each one as it occurs. This approach has the effect of compressing the best and worst cases into a guaranteed set of a test trials, and thus has problems in cases where $l \neq a$. If $l < a$, the a test trials is definitely worse than the best case above of l trials, and depending on the values of l and a may be worse than many other cases as well. If $l > a$, this system has the problem of assigning the additional labels; they could either be multiply assigned on each demonstration (resulting in an improvement on the former best case scenario of l trials), or the robot could run through its entire set of actions multiple times until all l labels have been assigned, resulting in a worst case trial set of as sets where s is the maximum number of synonyms in l. Either way, this hardly represents a natural interaction in terms of the way humans generally teach.

One further optimization is made in the label teaching process. The introduction of the concept of style axes implies bidirectionality: styles may have semantic opposites. For example, if traversing a style axis in one direction corresponds to increasing "fluidity", traversing it in the other direction might correspond to increasing "jerkiness". The symbol grounding hypothesis system has therefore been designed to incorporate the notion of opposites, in order to allow the human to ground symbols in terms of existing symbols, in addition to grounding them in the underlying symbolic adverbs directly. This can take place atomically, with statements like:

Teacher: *[label2]* is the opposite of *[label1]*.

Or as part of corrective feedback, with statements like:

Teacher: "Robot, do that action more *[label1]*."

Robot: *[performs the action incorrectly]*

Teacher: "No, robot, that's *[label2]*; it's the opposite of *[label1]*".

This now allows the teacher to specify two labels for every adverb trial action that is part of a given style axis. If the number of pairs of antonyms in l is p, the best case scenario in which all l labels are used is $l - p$ trials. However now the worst case scenario is simply $\max(l, a) - p$ trials — the range of necessary user involvement in the grounding process has been greatly compressed. Of course, the human can make mistakes, and the value of l can vary from interaction to interaction, but clearly this method is relatively efficient while maintaining the natural interaction flow of human instruction.

One final note is related to the comment above about the human making mistakes. Mistaken labelings can of course be corrected by the appropriate use of feedback, but a more common occurrence is not necessarily an error, but simply the omission of feedback once the robot has performed the trial action. It is generally reasonable to assume that the absence of feedback implies correct behavior on the part of the robot, because the human does not necessarily have the patience to explicitly reward the robot, particularly if he or she simply assumes that the robot knew what to do and is not looking for confirmation. Therefore, the absence of feedback is treated as a weak positive reinforcement, and explicit positive or negative feedback has a proportionally stronger effect on the value of the hypothesis support.

7.3.2.2 Implementation of the Style Axis Model

As mentioned in Section 7.3.2, the style axis model is designed to present a consistent interface from the human's point of view. Underneath this level, there are four main components: *labeled adverb converters*, the *action repertoire*, *overlay actions*, and the *speed manager*.

The general use of adverb converters for producing animation blends has been covered in Section 7.3.1, so there is no need to go into further detail concerning how the underlying motion alteration works. In order to add the style grounding capabilities of Section 7.3.2.1, the standard adverb converter was extended into a labeled adverb converter. This is actually a combination of an adverb converter with labeling capabilities, and a special action container that knows how to pass labels along to the adverb converter.

The motions that are loaded into the adverb converter are arranged along a logical set of style axes by the designer, and those axes are given internal symbols that are loaded into the label hypothesis system as the adverbs. For example, a 2-dimensional set of style axes can be set up to represent the robot's facial expressions, such that a range of emotions can be selected according to coördinates along the two style axes. Opposite directions on the style axes represent opposite emotions, as shown in Figure 7-17.

When a style alteration is requested for an action that is ultimately represented by a labeled adverb converter, this will consist of a label, an axis direction (i.e. "more" or "less") and a relative change amount. The label is passed via the action to the label hypothesis system. This system returns an internal adverb symbol, which is passed by the action, along with the axis direction and amount, to the labeled adverb converter, producing an array of blend weights in much the same way as a pair of numeric coördinates would ordinarily be converted into blend weights by a regular adverb converter. Because the weights are thus encoded in the action, in the form of the label and the axis direction and amount, the style change to the action is naturally persistent without any change to the underlying motions that comprise the adverbal blend.

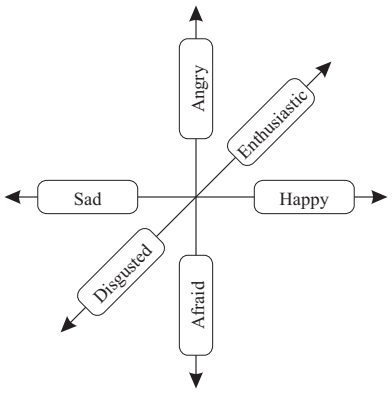


Figure 7-17: A theoretical example of style axes aligned in a traditional Cartesian space. Blending between emotions arranged in this way maps directly to the specification of a point using a separate coördinate along each logical axis.

Arbitrary style axes that can not be implemented through direct motor blends are handled by the action repertoire subsystem. Unblended motions within a particular motor system (typically divided up by bodily region) are arranged into a fully connected verb graph, to enable any motion transition to take place. A separate generic action is then instantiated for each motion, in the action system, but they are not yet set into fixed action-tuples with trigger percepts. Instead, these actions are all loaded into the action repertoire, which is essentially a database of all of the robot's motor skills.

This database performs a number of support functions for the teaching process, most of which are described in Section 7.4.3.2. Here the discussion will be limited to the support that the action repertoire provides for style axes. Each action that is loaded into the action repertoire is also tagged with two kinds of style metadata. The first set of metadata identifies the action classes that the action belongs to — this class identifier is a subjective content summary, such as "point" or "wave". The second set of metadata is the list of internal style axis symbols which are appropriate to the action — i.e., which style symbols might be used to convert this action into another member of its class — and the corresponding linear weightings that represent the action's positions on those style axes. Thus the stylistic contents of the action repertoire can be thought of as a loose collection of directed acyclic subgraphs, where each subgraph within a class represents a style axis (Figure 7-18).

The action repertoire also has an interface to the style label hypothesis system, and style label reference proceeds similarly to the way in which it occurs in the labeled adverb converter. When a style adjustment is applied to a specific action, the label, axis direction and amount are passed to the action repertoire, which looks up the prototype for the action in question and the internal style axis symbol for the label. The action repertoire then looks up all action prototypes that match the class tag of the action and that have positions on the same style axis. The closest action to the desired style axis position is selected, and a copy of it is returned via the action duplication interface (Section 7.4.3.2) to ensure that style changes are only persistent within the specific action instantiation, not within the prototype for that action.

One of the style axes that is always available regardless of the layout of the robot's motor system is speed. Although speed also has a direct effect on timing, it is considered to be part of the style system (many other style changes also affect timing, in any case). Alterations to motion

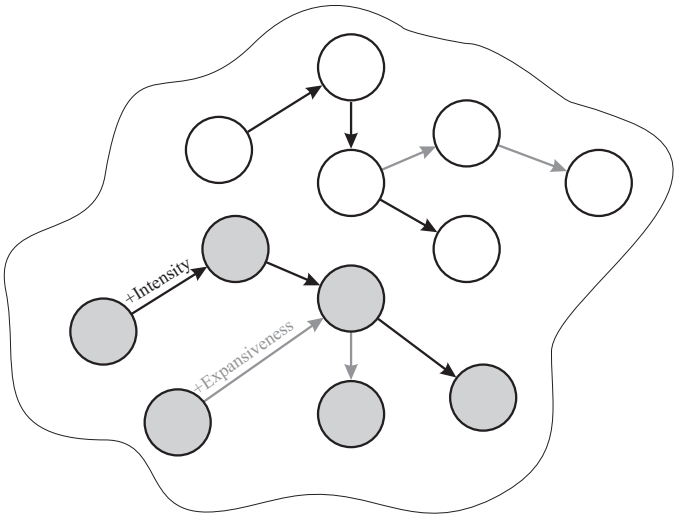


Figure 7-18: An example of arbitrary non-Cartesian style axes applied to actions in the robot's repertoire (the circular nodes represent actions, and the edges represent transitions along a particular axis). A given style axis is represented as connections between stylistically distinct actions of a given class, in a directed acyclic subgraph of the action repertoire. In this illustration, the two different action classes are represented by the shading of the nodes, and the two different style axes are represented by the shading of the edge lines.

speed had to be handled by a special subsystem, due to the fact that speed adjustment occurs by resampling the actual joint-angle series motion representation, rather than just adjusting an action that triggers the motion. Thus any speed adjustment to a motion is persistent across all actions — for example, speeding up a particular arm flourish in the sequence would otherwise result in all such arm flourishes, whether occurring before or after in the sequence, would be changed in the same way.

Speed adjustment is therefore handled by a module called the speed manager. One fortunate consequence of using explicit joint-angle series to represent motion is that it ensures that the robot can not be attempting to perform more than one instantiation of a given motion at any particular time. The speed manager's job, then, is to preserve a snapshot of the motion's original sampling rate, and swap in and out speed-adjusted versions of the motion when required (Figure 7-19). When an action requests a speed-adjusted version of a motion, it first contacts the speed manager to copy the motion's old rate, and notify the action of a weight that represents that action's desired speed constant. The speed manager then notifies the motor system to resample the motion, and the action is allowed to trigger the motion as normal. The speed manager keeps track of the execution of the motion, and when it finishes, swaps the original motion rate back in via the motor system. Thus once again, style adjustments can appear to be persistent across their individual action wrappers, while the underlying motions remain effectively static.

Finally, the way that the robot's motor systems are typically divided up into multiple motor systems according to body region means that there needs to be a mechanism for combining style adjustments across multiple areas of the robot's body. For example, consider the situation when the robot is being taught to make an obscene hand gesture, and the human decides that it should be accompanied by an angry facial expression. One way to accomplish this is of course to define two separate actions, the hand action and the facial action, and synchronize them to the *equals*

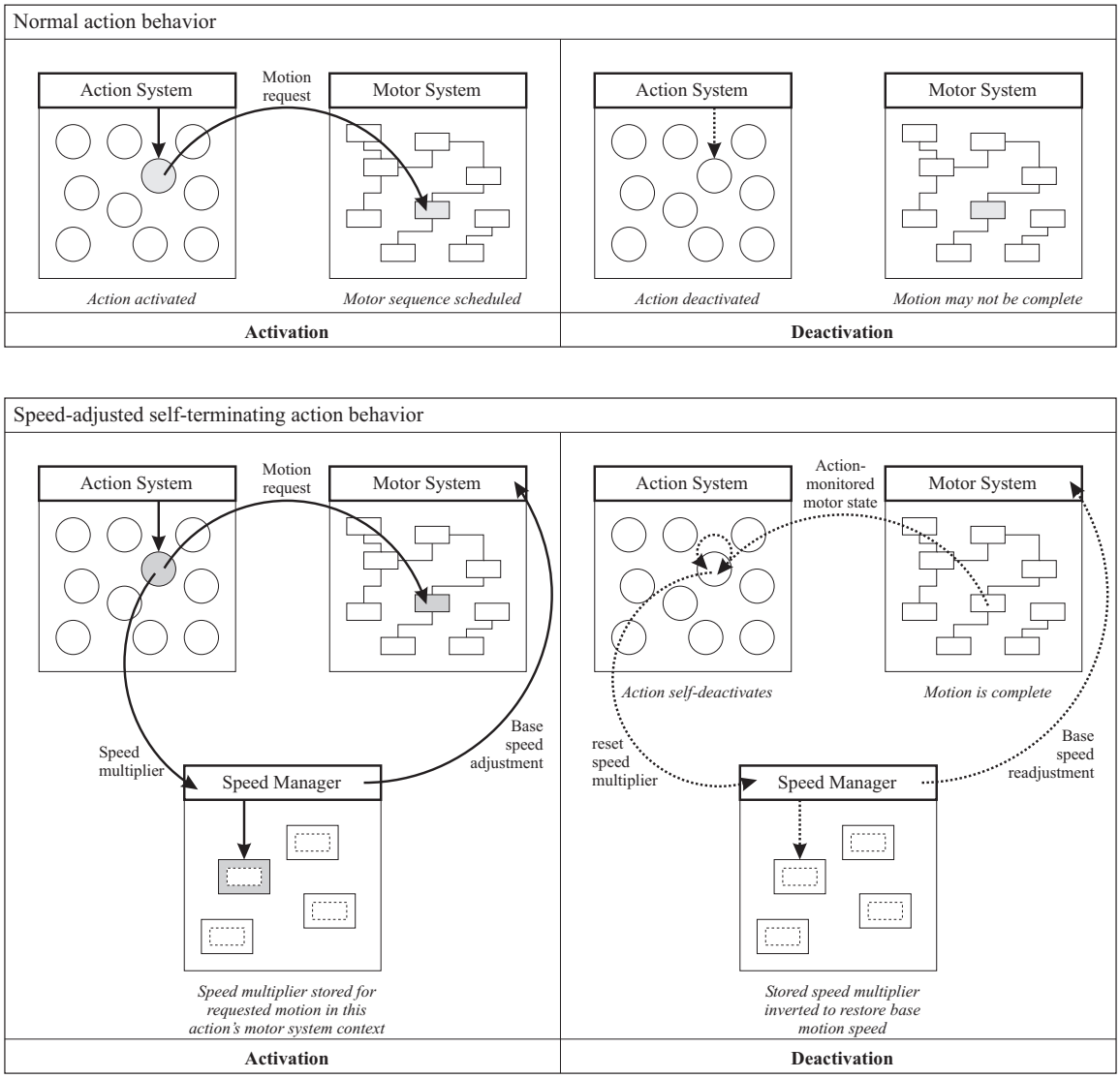


Figure 7-19: Illustration of how the speed manager mediates between actions and the motor system in order to perform just-in-time adjustment of the speed of a motion for an instantiation of an action, without allowing the change to persist beyond the activation of the action. The action itself behaves largely as usual, with the exception that it must be a self-terminating action in order to prevent premature deactivation of the action from triggering speed modification of a motion in progress.

relationship using the timing system. This method can be used, but since in this case the facial accompaniment is essentially a stylistic adjustment for the hand gesture, it is neater to have it more tightly attached to the gesture itself. Consider the situation where at a later date the human decides to have the robot change its expression to a smile, to indicate that the gesture is made in jest — in this case it might make logical sense for the human to apply the style change to the gesture, not the facial expression directly.

Here is where the overlay concept is brought in again. All of the actions in the action repertoire are special overlay actions, that have links to other actions that take place synchronously. This arrangement is conceptually similar to the timed actions mentioned in Section 7.2.2.1, that keep track of estimates of their own duration, in the sense that it provides adjunct functionality to the core action encapsulation. When an overlay action is triggered, their overlays are triggered simultaneously, and they are similarly detriggered together. Thus the overlays are intended solely to contain motions that have easily variable durations, primarily facial emotions and simple body postures.

Since the overlays are actions themselves, they are subject to the existing style axis implementations covering the way their motion control is handled. For the most part, the overlays are likely to be controlled by labeled adverb converters, since facial expressions and simple postures tend to be easily arranged into blend groups.

This concludes the implementation details of the style axis model. In the following section, I explain how all of these elements are put together into a process of interactively teaching and shaping an action sequence on our expressive humanoid robot, "Leonardo".

7.4 Sequence Assembly and Shaping: Teaching a Robot Actor

The process of interactive assembly and shaping of a communicative sequence will be illustrated using an application whose relation to the subject is clear if not even obvious: teaching a humanoid robot to act out a dramatic scene. Dramatic acting is a prime example of subtle and expressive communication. The style of a performance is as important as the raw components of movement and dialogue; this is why the choice of an individual actor, with his or her accompanying traits of style, personality and approach to the part, is such a critical factor in the dramatic arts.

Animatronic machines have featured as actors in theatre and film for some time, but the use of the word "actor" to describe them is somewhat generous; they are typically puppeteered by humans out of view of the audience, so robots that react autonomously to other actors or the audience are rare (see Section 4.3 for an introduction to interactive robot theatre). Robots that autonomously exhibit dramatic acting skills are rare; one example is the work of Bruce et al., which looked at the skill of improvisation. In this research, mobile robots were developed to improvise acting dialog among one another, in order to convey an impression of believable agency[53]. However in this case we are more interested in using interaction to refine the performance of a robot acting with a human: to facilitate "directing " the robot, in the dramatic sense.

The work in interactive authoring of character performance has already been summarized in Section 7.1.1, so I will here only mention some relevant related work that applies directly to dramatic acting — this particular area has so far received only thin attention. Tomlinson has proposed using human-robot acting performance as a benchmarking technique for evaluating the empathic

ability of what he refers to as "heterogeneous characters", meaning synthetic actors that are robotic, animated, or a combination of both[286]. Rather than evaluating the character directly, his proposal involves conditions in which human actors act an identical scene with either a heterogeneous character or a human control, and blind evaluation of just the human subjects' performance alone is used to infer the empathic skills of his or her co-actor. I fully support this method as an evaluation technique, however I am concerned with the process prior to this, in which the robot is taught the part itself.

Performance refinement of dramatic acting has received some attention within the Virtual Reality (VR) community. Slater et al. detail an experiment involving a non-immersive VR that was designed to allow pairs of human actors to rehearse a part under the guidance of a human director, all of whom were otherwise remotely located[272]. The actors were represented in the VR by virtual avatars, and could control a number of visual aspects of their avatars such as facial expression (via a palette on the screen), a few body postures, and navigation around the rehearsal space. The four-day experiment was evaluated using questionnaires, whose results showed that over the time frame of the experiment the participants felt increases in their feelings of presence, coöperation, and the amount that the situation resembled a "real" rehearsal. The subjects generally agreed that the experience was superior to both solitary learning of lines and collaborative videoconferencing for generating a plausible performance. These results reflect positively on the prospects for this application of interactively directing and rehearsing with a situated robot.

A final note on the interactive shaping process as it relates to theories of dramatic acting. Acting is a talent that aims to communicate through portraying a convincing impression on the part of the actor. How the actor should mentally accomplish this is a subject of preference in the world of drama. Traditionalist acting schools have asserted that actors should not cloud their performances with subjectivity by attempting to consciously experience the feelings they are portraying, but instead act "from thought": deliberately and detachedly imitating the feelings rather than personally assuming them, as summarized by Diderot[84]. Clearly, since interactive shaping involves no manipulation of the robot's internal emotional or motivational state, it can be equated with this school of drama.

Other schools of acting, however, take a contrary approach. Method acting, pioneered by Stanislavsky, instead dictates that a performance can appear most genuine if the actors strive to personally experience and channel the desired feelings at the moment of portrayal[70]. I do not intend to support one dramatic school of thought over another, and indeed I believe that it is most advantageous for a robotic actor to be able to make use of both. The corresponding analog of method acting, the manipulation of internal emotions, drives and personality, was alluded to in Section 6.3 on support data for behavioral overlays. In this material I focus instead on the deliberate approach.

7.4.1 Progressive Scene Rehearsal

The process of teaching Leonardo (Leo) the scene is one of progressive, collaborative refinement: the human and the robot work through the scene like a rehearsal, going from an initial coarse arrangement to increasingly fine adjustments. During this process, the robot essentially moves among four distinct modes: *Assembly*, *Shaping*, *Imagining*, and *Replay*.

These modes have been chosen primarily to manage the human's expectations of what the robot is able to do at any given point in the interaction. A distinction has been drawn between the addition of atomic actions to the sequence in the Assembly phase, and the manipulation of those actions in the Shaping phase, because there is some ambiguity in the use of language to select actions in the robot's action repertoire for inclusion in the sequence versus selecting them within the sequence itself for modification. Rather than insist on constraining the human's speech grammar with unnecessary features to distinguish between these cases, the separation of the two phases allows the robot to have different default command responses.

The Imagining phase is a somewhat novel state in terms of human-robot interaction — while a state of contemplation, reflection or even "dreaming" has been suggested numerous times within the field for allowing the robot to refine its task performance without direct human intervention, I am not aware of any instances in which it has been implemented so explicitly. Again, the primary reason for its inclusion as a separate phase is to ensure that the human understands that the robot is occupied and cannot immediately respond to requests to demonstrate its part.

Similarly, the Replay phase was designed so that the human can be assured of being able to act out the sequence along with the robot without worrying that any adjunct dialogue will be misinterpreted as a shaping command. Descriptions of each of these phases follow.

In the Assembly phase, the human's goal is to teach Leo the envelope of his performance. This consists mainly of selecting the actions that the robot needs to perform; timing considerations are mostly qualitative, concerning the general synchronization and sequential arrangement of the actions. Similarly, stylistic considerations are also mostly dispensed with at this point, other than gross emotional and postural actions.

Mechanistically, the Assembly phase consists of adding actions to Leo's internal Scheduled Action Groups. As actions are added to a Scheduled Action Group, it keeps track of a "reference action" that represents a current context pointer within the timing hierarchy. This aids in the construction of a reasonable hierarchy, as new actions can be added as the children of the reference action if they are not explicitly synchronized as dependents of a specific action. An illustration of the Assembly process is shown in Figure 7-20.

The Assembly procedure is a characteristic example of the process of "learning by doing". Each time the human requests an action to be added to the scene, Leo acknowledges his understanding of the request with a nod and demonstrates the action that he has just added to his Scheduled Action Group. This provides an opportunity for the human and the robot to maintain common ground — the human thus sees exactly what Leo's understanding of the request was — and for the human to provide feedback. If Leo performs an incorrect action, the human can verbally point out his error immediately, and Leo will remove the offending action from his sequence.

Similarly, if at any time during the process, Leo does not understand what the human is requesting, or is unable to comply with the request, he can communicate this fact back to the human. Leo does not speak, so he accomplishes this with symbolic bodily gestures. Leo signals his confusion — for example, if the human makes an ambiguous or malformed request — with a shoulder-shrugging action. Conversely, if Leo concludes that he has correctly processed the request, but is unable to perform it, he will signal his refusal with a head-shaking action.

Once the sequence has been populated with actions, the human tells Leo it's time to "work on" his part, giving him the signal to move into the Shaping phase. The qualitative arrangement

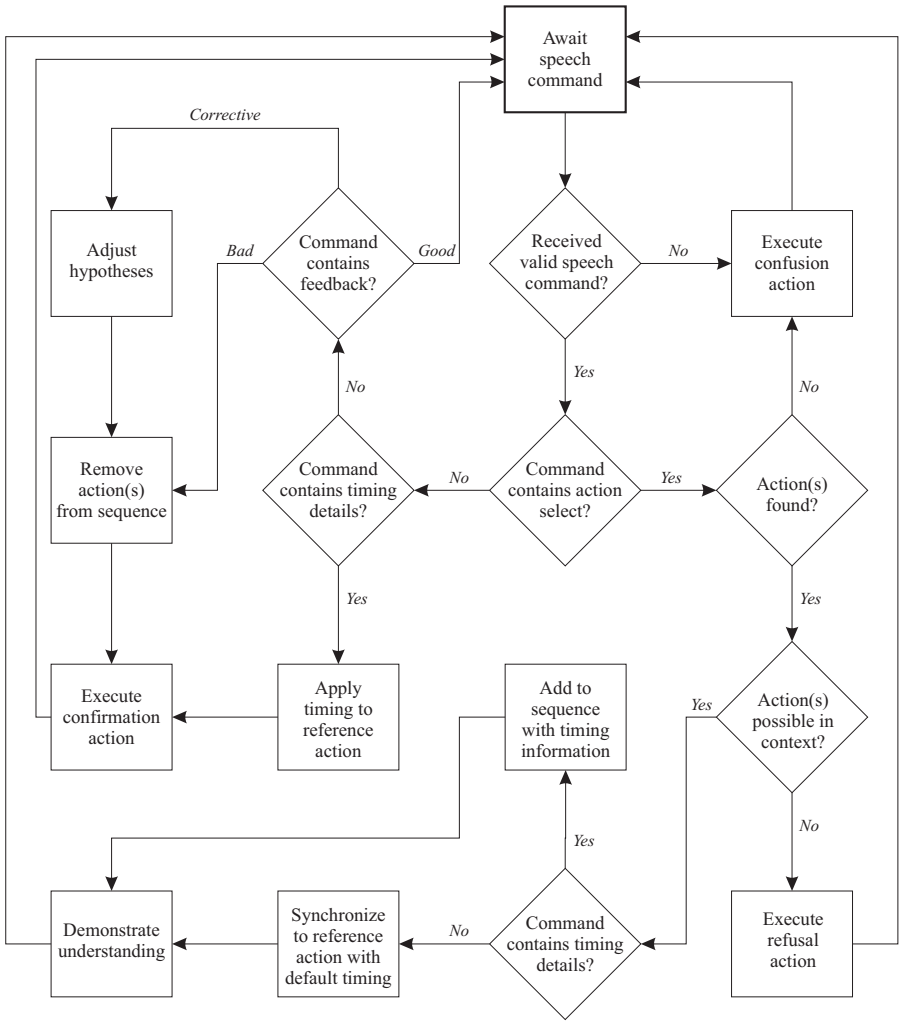


Figure 7-20: Flow overview of the initial Assembly phase of the interactive sequence teaching and shaping process.

of the Scheduled Action Groups is decomposed into a precise schedule, and the robot is ready to begin fine-tuning the timing and style of the scene. Shaping essentially proceeds as a process of selection and adjustment: the specific atomic actions to be shaped are referenced in various ways such that the human and the robot are jointly attending to them, and then particular aspects of the reference action can be modified.

To understand how actions are selected, first consider how a human director would coach a human actor. The primary means of establishing joint reference is for the actor to rehearse the scene, and the director to stop the rehearsal at points that needed adjustment. The director can also explain to the actor which actions needed alteration, through a description. Finally, the director can get up and demonstrate to the actor how he or she wants a section of the scene done — the actor recognizes the section from its component actions, and compares the director's demonstration with his or her own memory, extracting the important differences and resolving to apply them in the future.

The interactive shaping of Leo's acting rehearsal similarly makes use of these three action reference modalities: temporal, verbal, and physical, in this order. Since Leo has generated a preliminary schedule, the shaping interaction proceeds as a rehearsal, with Leo "running" or "stepping" through the scene according to the human director's commands (e.g. "Action!" and "Cut!" from the motion picture world). If the director stops Leo's rehearsal, the robot has a temporal locality of reference — the context of the director's next instruction is likely to concern one of the actions it has just performed.

The human director can then use the verbal modality both to further disambiguate the reference action and to apply the shaping, through timing commands and symbolic style adjustment statements (Sections 7.2 and 7.3). Disambiguation occurs as necessary; if Leo is not able to sufficiently disambiguate the reference action, he will shrug to ensure that the human understands that more specificity is needed. If the temporal context of the action is unambiguous, the human can use a deictic phrase to refer to it, e.g. "Do that more slowly". Otherwise, a slightly more qualified deictic phrase can be used, incorporating a body region to isolate the action referent (see Section 7.4.3.2 for more details), e.g. "Move your hand more slowly". Finally, if all other verbal disambiguation mechanisms are insufficient, the human director can exactly reference an action according to its cardinality within the sequence, e.g. "Do your first hand movement more slowly".

Of course, physical demonstration is often a more natural way of precisely referencing actions than linguistic description. Unfortunately, the scenario described above, in which the robot automatically extracts and applies the important part of the demonstration, is a major unsolved problem in imitation learning, and I have not solved it either. Nevertheless, I have attempted to incorporate two limited elements of it in the interactive shaping system. First, the human can reference certain actions in the sequence by performing them while informing the robot that it is being given a demonstration. Second, the human can adjust the timing of certain actions by demonstrating their desired relative timing through real-time expression of their start points.

The robot maintains a fully articulated "imaginary" internal model of itself, and when it is not being used for other mental speculations it acts as a "mirror neuron" interpretation system, mapping the human's posture onto the robot as though the robot itself was doing what the human is doing (Sections 2.2 and 3.3.2). The human's posture is captured with the use of a Vicon motion capture rig. When the human performs a demonstration for action reference, the robot compares the motion of its imaginary body with the pre-analyzed stored motion sequences from its initial repertoire using the dynamic warping technique described in Section 3.3.1. The best match that exists in the current sequence is then selected as the reference action.

Similarly, as the robot runs through its rehearsal, it keeps track of a current time pointer within the schedule as well as the temporal reference action. The human can use physical and verbal demonstrative cues to cause the robot to move selected actions to the location of the time pointer in absolute time. However, the absolute time schedule is not canon: it is frequently recomputed by decomposition of the interval-based action time representation (Section 7.2.2.1). Therefore, the new location of the action must be converted back into a set of qualitative interval relationships and, if necessary, an associated spacer interval. This is accomplished via the network rearrangement technique presented in Section 7.2.2.2.

An illustration of the Shaping phase is shown in Figure 7-21. One aspect must be mentioned in more detail. When an action is shaped, the robot plays it back so that the human can monitor and evaluate the change and give feedback where appropriate. However, simply playing back the

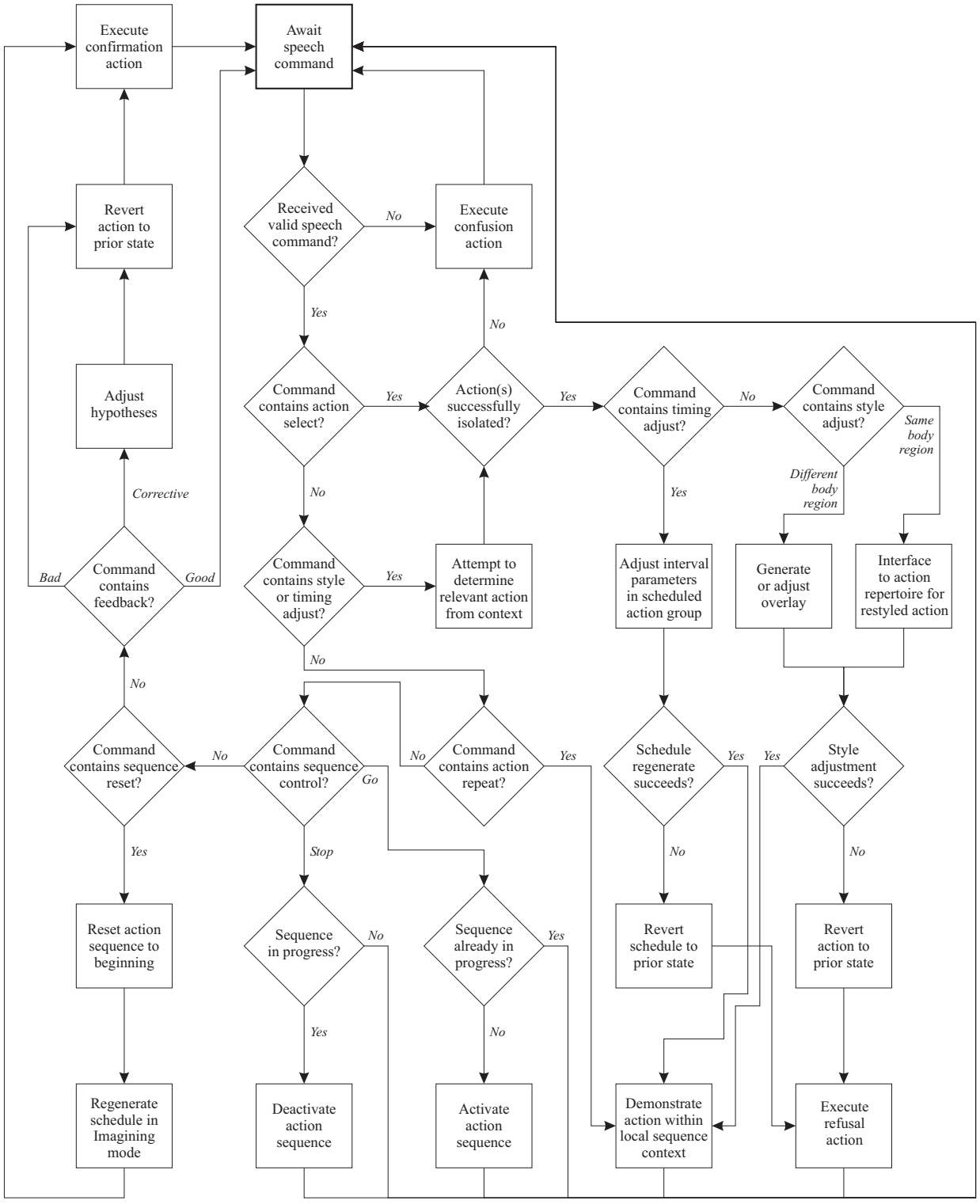


Figure 7-21: Flow overview of the Shaping phase of the interactive sequence teaching and manipulation process.

action without its surrounding context does not give the human an accurate impression of how the style of the action has changed, and certainly not the timing as this is relative to other actions. Therefore the robot uses the temporal locale of the reference action as well as the hierarchy of the nearby action subnetwork to select a set of actions that are likely to be relevant to the one that has just been shaped. It then executes just this set of actions, in the relative timing specified by the schedule, as its response to the human so that he or she can form an accurate contextual opinion of the validity of the changes that the robot has made.

Earlier I referred to other "mental speculations" that the robot performs using its mental body model, which I term "imaginary Leo". This primarily refers to a process of mental temporal rehearsal which Leo performs in order to accurately decompose the action timing network into a precise schedule (see Section 7.2.2.1). This occurs whenever Leo completes an Assembly or Shaping phase — before entering whichever destination phase has been requested, he goes through a phase of Imagining. During this mode, he executes his action sequence on his imaginary Leo. Because the imaginary model behaves identically to the physical robot, he can monitor the duration of action components to a high precision.

Of course, this also means that the full Imagining process takes exactly the same elapsed time as a physical rehearsal. In order to prevent the human from experiencing tedium, the robot skips over actions whose durations are known not to have changed. While the robot is in its Imagining state, its physical body performs a contemplative action so that the human knows it is cognitively occupied and has not yet entered its destination mode. The Imagining phase is depicted graphically in Figure 7-22.

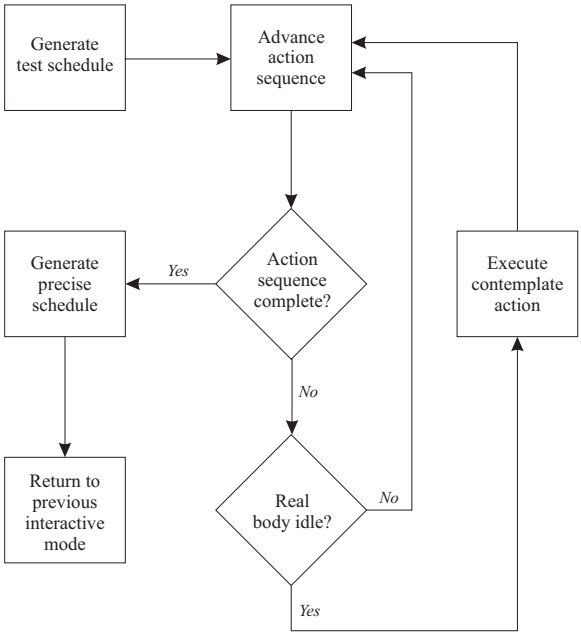


Figure 7-22: Flow overview of the Imagining phase of the interactive sequence teaching and shaping process.

The final mode, Replay, is simply a lightweight version of the Shaping mode that does not contain the functionality for action reference and adjustment. It is intended as the performance mode for the acted scene, in which the human is just another actor rather than the director. This

allows the human to deliver his or her cue lines without the potential for confusing the robot into thinking a shaping request is being made. Since timing cannot change during Replay, there is no need for the robot to return to the Imagining state when the performance concludes. The robot will remain in the Replay state after executing the sequence, in case the human wants to back up and rehearse it again; good feedback on its performance will cause it to acknowledge and exit to a neutral mode.

7.4.2 Decomposition of Speech and Demonstration

The priority of this interaction is its naturalness for the human; ultimately the same results could be achieved with offline animation or programming, but the goal is to empower human-style acting performance shaping. Therefore the speech and demonstration systems have been designed to try and avoid restrictions on the human's behavior.

Humans have a large vocabulary of words, but the conceptual vocabulary of the acting domain is smaller. I therefore use a tagged grammar technique to perform a progressive vocabulary condensation, similar to the technique used in Section 5.3.3. This can essentially be summarized as front-loading the verbal vocabulary onto the speech recognizer, which is most equipped to handle it, and arranging the grammar in such a way as to be able to extract the important conceptual vocabulary to pass on to the robot's behavior engine.

This is accomplished with a two-stage parsing process. First, tags are added to the grammar in a carefully defined order, so that as the tags are activated during the speech parse they form a string that can then itself be parsed in a second stage into simple commands. In other words, when a given string of human speech is parsed by the speech recognizer, a string of tags is generated that forms a sentence in the acting vocabulary. This is sent to a perceptual preprocessor in the robot's behavior engine. This preprocessor uses string pattern matching to chop up its input string into a set of discrete command items. These command items are then packaged into separate command percepts, which are then all sent to the robot's perception system to be perceived as incoming speech commands.

What actual commands the robot will perceive depends on the composition of this set; commands can reference one another, and this interface also allows the robot to receive multiple commands within a single utterance on the part of the human. This represents an extension of Leonardo's earlier speech command processing system, which assumed all utterances to be single commands with unique referents. The types of extended speech commands that the robot can now process are as follows:

- *General Speech Command*: encapsulates the second-stage parsed speech phrase as a string. Simple commands such as mode changes, feedback, and start and stop commands can be represented simply with this type of command structure without additional specialization.
- *Timing Adjust*: encapsulates an adjustment to the timing of one or more concurrently specified actions. Timing can be modified in an absolute or relative sense. The command also contains flags for specialized timing adjustments: synchronization between actions, synchronization to the current point in the running timeline, and synchronization to a speech cue in the acting sequence.

- *Style Adjust*: encapsulates a style axis traversal on an explicitly or implicitly specified action. The command contains the style label in question, and a modifier (all style axis modifications are expressed in a relative sense).
- *Body Adjust*: refers to the addition of an action to the sequence that alters the robot's body configuration (thus an action selection from the action repertoire, rather than the current sequence). The command's optional parameters are body region tags, a modifier, a left/right symmetry disambiguator, a flag signaling a concurrent physical demonstration, and specific physical configuration descriptors that can be provided by the designer.
- *Action Select*: refers to selection of actions that can take place either within the action repertoire or the current sequence, depending on the robot's acting mode. The command's optional parameters are body region tags and a sequential ordering number for indexing into the action repertoire or the current sequence.

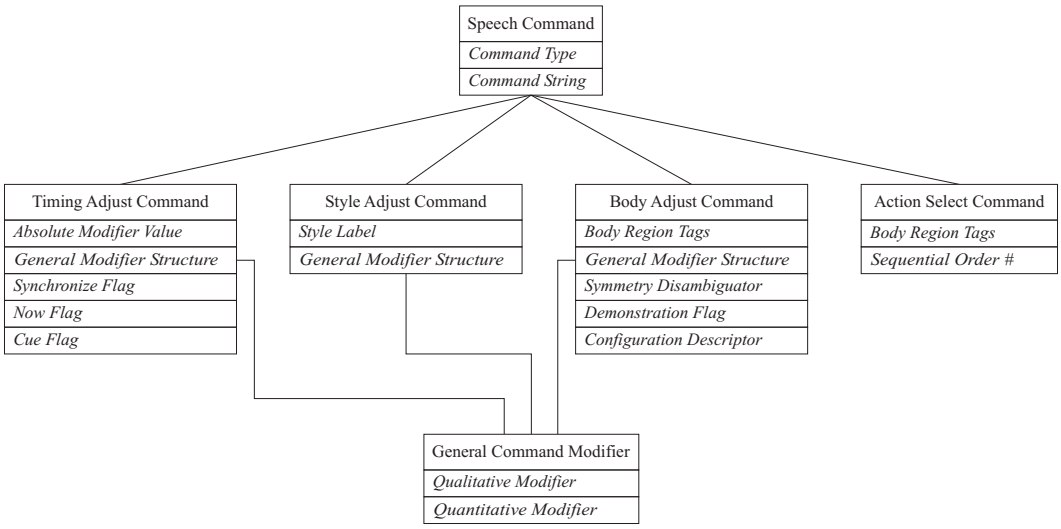


Figure 7-23: Structure of the robot's speech command set, extracted from the two-stage grammar parse. Specific speech commands are descended from a single parent class, and several may contain a link to an associated command modifier class.

As illustrated in Figure 7-23, the different commands are descended from the parent speech command class which preserves the second level speech string. The command modifier structure, an instantiation of which several commands are able to reference, is a general mechanism for encoding relative modifications. The qualitative modifier encodes the direction of modification (e.g. increase or decrease), while the quantitative modifier refers to the amount of change desired.

An example parse of an incoming speech phrase is shown in Figure 7-24. The sentence is first parsed via the Sphinx grammar (relevant fragments only shown — see Appendix A for full grammar) into a second stage vocabulary string. This string is then re-parsed by a cascade of pattern matchers into a collection of speech commands. Each speech command is then encapsulated into a separate `C5M` percept and released to the perception system.

Currently the scope of content variability of some of these commands, particularly style adjustment, is limited because the set of tags that are embedded into the speech grammar is static.

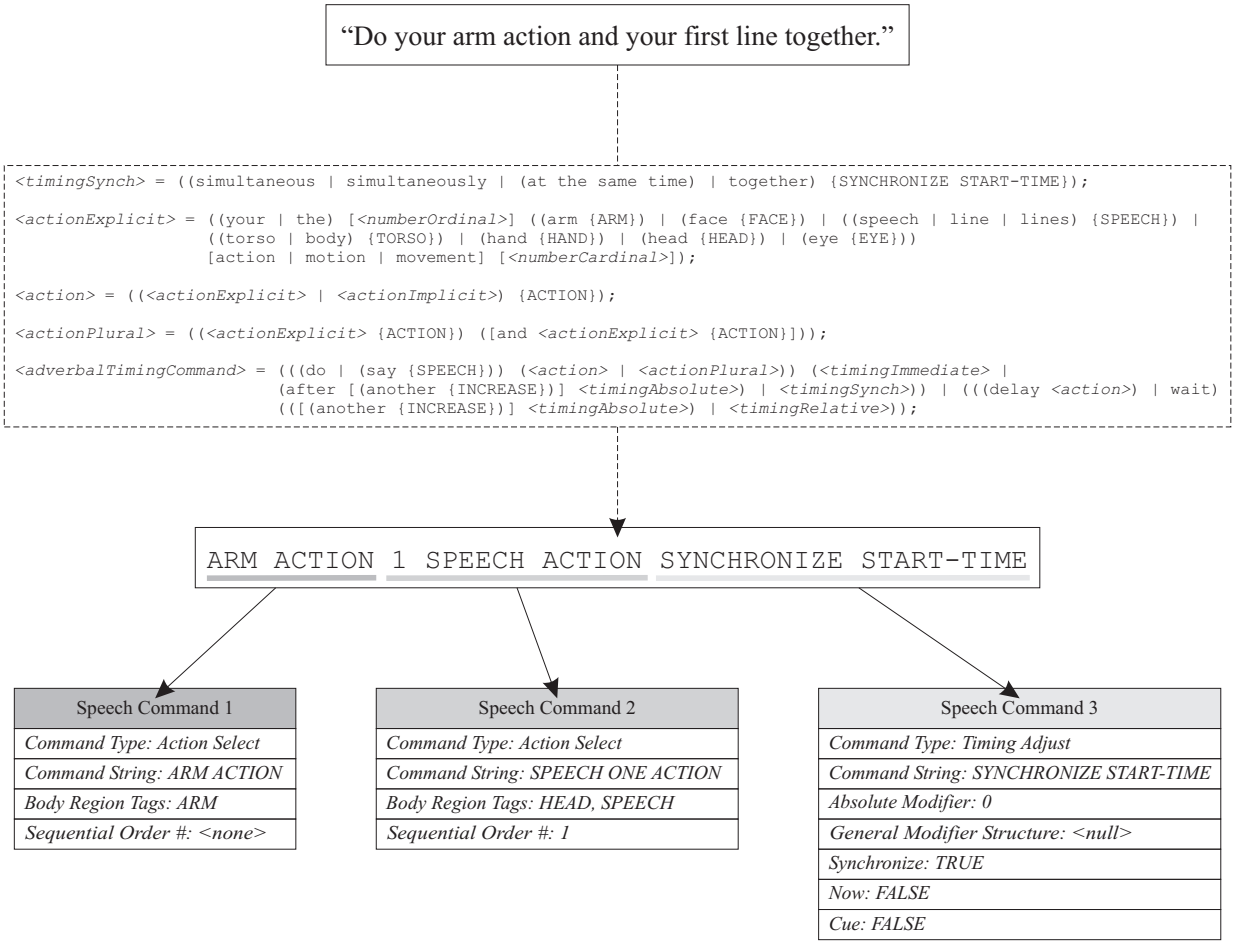


Figure 7-24: Two-stage parse of an example speech phrase, showing the decomposition of the phrase into an intermediate string and its subsequent conversion into individual speech commands.

Ideally, it would be possible to set dynamic tags that extracted the word at that point in the speech parse and converted it to a tag to be inserted into the condensed vocabulary string. At present, however, we do not have that functionality, so the tags have been designed to provide a basic coverage of stylistic labels. When this functionality becomes available, it can be easily incorporated as the style symbol grounding procedure has been explicitly designed to accommodate arbitrary labeling.

The decomposition of continuous human motion into segments for action reference is also accomplished using speech, in the form of transmodal bracketing similar to that used to extract dynamic deictic gestures in Section 5.2.2. Although the research concerning the synchronization and overlap of speech and gesture phrases only specifically concerned deictic gestures, it is assumed that the overlap will be substantial for other gestures as well. This assumption may not be valid in a large number of cases, however the ability to specify reference actions in this system is already highly redundant through the use of temporal, verbal and physical referencing. If physical disambiguation does fail in some cases, the human has multiple opportunities for recourse.

The arrival of a physical action reference command, therefore, is treated as the end of a spoken

phrase which is assumed to correspond to a point in the gesture phrase that is at or after the nucleus. When the imaginary Leo is not being used to mentally rehearse timing, and is thus operating as a mirror neuron system for the human's activities instead, it keeps a buffer of the mirrored pose as a kind of postural short-term memory. The length of this buffer is sufficient to encompass any of the robot's prototype motions, the durations of all of which are known when they are preprocessed during the robot's startup procedure. The most recent section of the speech-delimited postural short-term memory can then be compared with all relevant prototype motions using the technique described in Section 3.3.1.

This forms a very rudimentary action recognition system that is heavily dependent on the constraints on the set of motions to be matched against. Indeed, one of the advantages of the interactive teaching mechanism is that techniques such as this, that would otherwise be completely unscalable, become useful due to the scaffolding inherent in the socially guided tutelage approach.

7.4.3 Action Generation, Selection and Encapsulation

As explained in Section 7.3.1, the majority of the atomic actions that are sequenced in the interactive shaping process are prefabricated encapsulations for animated motions that have been created in advance, or interfaces to direct joint motor systems. The former have been thoroughly described. An example of the latter is the set of reflex actions that control the robot's movements in reaction to touch stimuli (Section 3.2).

However there are a few classes of actions that are dynamically generated by the sequencing system, rather than referring directly to single joint-angle series animation motions. These are lip synchronization actions for realistic dialogue acting, self-imitating postural actions for assuming specific poses, continuous autonomous behaviors incorporating multiple motions, and mimicry motions generated from direct imitation of the robot's mirror neuron mapping. These are described in the following section.

The selection of reference actions in the shaping process once the sequence is assembled has been described earlier in this chapter, but there is more to be said about how actions are initially selected by the robot in response to human commands during the Assembly phase. In addition, there are several interfaces beyond the standard action-tuple metaphor that actions in this system must provide in order to support the shaping process. This is mostly presented in Section 7.4.3.2, other than certain aspects of the Action Repertoire that were already covered in Section 7.3.2.2.

Finally, there are a few other action interfaces and encapsulations that have been alluded to but not fully described, in particular those relating to the use of turn-taking boundaries to manage the size of Scheduled Action Groups. These remaining details are provided in Section 7.4.3.3.

7.4.3.1 Dynamically Generated Actions

There are several cases in which it is necessary or advantageous to generate atomic actions dynamically. These cases are essentially covered by two conditions: when even unshaped atomic actions cannot be predicted in advance (such as learned actions), and when actions are so amorphous as to make their explicit prior representation onerous or absurd. Some of the actions below share overlap between these two cases.

Although Leonardo does not yet have a speech generation capability, for the acting application it is important for him to be able to perform spoken dialogue. He has therefore been equipped to play back recorded speech as though he is speaking it, by performing lip synchronization movements in concert with the audio. Because this system actually generates his facial movement sequence in real time, rather than having the entire speech motion pre-animated, it is treated as a dynamically generated action.

Lip synchronization has a long history in the animation world, both for traditional cel-based animations and computer-animated characters[25, 226]. Originally, of course, animation sequences were created by hand, and consequently guidelines were developed for animators concerning the procedure. One such rule of thumb is that the less realistic the character, the less realistic the lip synchronization animations need to be in order to maintain the viewer's acceptance of its "visual shorthand"[180]. Leonardo is a robot designed for lifelike appearance, so it is therefore necessary to pay careful attention to his lip synchronization behavior to ensure that it conveys a natural and convincing impression. Breazeal gives a good introduction to the basic scenario of giving lip synchronization abilities to the more cartoon-like robot Kismet in [39].

In recent years, the availability to the animation industry of software-based techniques for automatic generation of lip synchronization poses has increased[171, 159]. These methods essentially rely on the practical reason behind the observation above concerning animation realism: the generation of speech sounds relies on precise control of mouth shape, but if the sound has already been made the mouth shape becomes just a visual adjunct that only has to correspond within a certain tolerance. There can thus be a compression mapping between the set of distinct sounds in human speech (about 45 for the English language) and the set of visually distinguishable mouth postures (18 are recommended for realistic animation)[227].

For Leonardo's lip synchronization behavior, we[3] use automated speech decomposition software and a set of hand-animated facial postures. The speech processing software is a commercial product, ProductionSync from Automatic Sync Technologies, aimed at providing lip-sync capabilities to the animation industry[15]. Following the recommendations in [227], we use 18 facial postures, plus a neutral pose to represent silence. Since humans do not move only their mouths when speaking, but synchronously actuate their entire faces — in fact, an animated character who speaks using only their lips and jaw looks extremely weird — our facial postures, or "visemes", use all of the robot's expressive facial actuators. Still images from all of Leo's visemes are shown in Figure 7-25.

The ProductionSync software takes as input an audio file and a text string containing the words being spoken. Therefore at present we must predict in advance the words of dialogue that Leonardo will speak. Ultimately, of course, automated speech recognition will progress to the point that this is unnecessary, but for the purposes of acting it is not an onerous requirement since a standard part of dramatic acting is knowing ones lines in advance. The software extracts phonemes from the speech and outputs an "X-Sheet" (exposure sheet, a standard reference device in the animation industry) timing representation. This timing information is fed into a *Viseme Adverb Converter* that reactively blends between the appropriate visemes as the X-Sheet timings occur.

Because most aspects of the lip synchronization motion are determined by the underlying audio file from which it is generated, such as the speed and relative emphasis of the spoken words, there is currently only one style axis available for the robot's speech actions. This style axis

[3]Leonardo's standalone lip synchronization behavior was developed by Allan Maymin. I incorporated the behavior into the robot's action system and made a few optimizations.

Figure 7-25: The 18 facial "viseme" postures generated as a basis set for Leonardo's lip synchronization behavior.

is internally given the symbol *intensity*, though of course the human teacher can refer to this axis by whatever label he or she desires, according to the style symbol grounding technique of Section 7.3.2.1. Since the actual motion output of lip synchronization is controlled by a continuous sequential adverb blend, intensity is adjusted simply by altering the proportion of the silent pose in the blend weights, thus restricting the degree to which the other visemes are expressed while preserving their occurrence.

Another set of dynamically generated actions are self-imitative postures. At first glance the notion of the robot imitating itself may seem absurd. However the technique of physically manipulating a learner through a task, so the learner learns from its own bodily experience, is common in some spheres of socially situated teaching behavior such as that exhibited by many primates (see Sections 3.1 and 3.2). In this case, if the human teacher wants the robot to adopt a certain posture in the acting sequence, the teacher can achieve this by somehow causing the robot to adopt the posture and then instructing the robot to do later what it is now doing. The robot thus plans to imitate its current activity at some point in the future.

Since the posture may not be predictable in advance, implementation of this behavior requires the encapsulation of arbitrary postures into a dynamically created action. In Leonardo's acting scenario, this is most commonly used to generate arbitrary hand poses for use in gestures. Humans use hand configurations for a variety of symbolic purposes, such as indicating integer quantities, or expressing sentiments ranging from "OK!" to "Go away!". It is convenient to be able to teach these with efficient self-imitation, rather than designing them all in advance. Using this method, the teaching process is as simple as manipulating the robot's hand into position and then issuing a verbal command such as "Hold your hand like that".

Of course, the robot has two hands, so there needs to be a mechanism for left-right disambiguation. This can easily be accomplished by issuing a more specific spoken command, e.g. "Hold your left hand like that". However it is not natural for humans to always articulate directions with such specificity, because the hand that is being referred to is usually clear from the interaction context. In this case, it is the hand that the human has just been manipulating. Contextual disambiguation is accomplished in this system with Leo's internal attention mechanism (Section 3.4).

Self-imitated postures also do not have many style axes, as they are dynamically generated based on actual behavior. If the teacher desires a different posture, which may reflect some different style, interactive alteration of the posture can be achieved using the same mechanism of compelling the robot to adopt the new posture and then self-imitating to create an associated action. At present the only style axis that has been implemented is the speed of transition into the posture, which is achieved by altering the velocity parameters of a direct joint motor system.

Low-level autonomous behaviors that may consist of multiple small reactive motions are difficult to encapsulate into actions in advance. One such behavior is Leonardo's autonomous maintenance of eye contact with the human teacher. Under ordinary interactive conditions, Leonardo attempts to follow social conventions by finding the human's face and visually attending to it, unless something else captures his visual attention.

This behavior is best treated as an abstract feedback-driven activity, rather than a specific sequence of conditions and associated motions. Nevertheless, it is desirable to be able to modify it, because the presence, absence and degree of eye contact is a powerful non-verbal communications channel (see Section 6.1.2). There consequently needs to be an action-based mechanism for deliberate manipulation of the behavior as a whole.

At present, Leonardo's eye contact behavior is not parameterized, such that it would be possible to convey style axes such as "detachment", "rapt attention" or "shiftiness". The system instead makes the best of things with binary control of eye contact. Actions can be dynamically generated that effectively switch Leo's autonomous eye contact behavior on and off at the will of the instructor. By judicious dynamic control of when Leo will and will not attempt to look at the human actor with whom he ultimately plays the scene, the instructor can convey a reasonable range of subjective impressions.

Finally, an ultimate goal of interactive instruction remains to develop mechanisms for teaching robots motor skills through observational imitation of an external demonstrator. Though generally an unsolved problem, support for direct external mimicry has been designed into the system. Since by definition motions learned through mimicry can not be predicted and encapsulated into actions in advance, this support also requires the use of dynamically generated actions.

Leo has the ability to mimic a human's body motions via his body mapping mirror system (Section 2.2), so rudimentary support for motion mimicry was straightforward to implement. The human directs the robot's attention to the appropriate body region and to the fact that an imitative demonstration is taking place — at present this is primarily accomplished with transmodal bracketing through speech — and the robot processes its body-mapped representation of what it observes in order to know how to replicate the motion via its own physical actuators.

Unfortunately, the current design of the `C5M` motion representation (described in Section 7.3.1) prevents externally imitated motions from being incorporated into the robot's behavior "on the fly". Recall that motions are represented as joint-angle time series, and that they must be resampled, checked for joint set parity and connected into the verb graph when the motor system in question is first created. Since there is currently no facility for doing this dynamically, this particular example of support for dynamic action generation has only been implemented at the action level and not at the motor level. Modifications to the motor system to accommodate this are planned, so the sequence assembly and shaping system supports them in anticipation of their completion.

7.4.3.2 The Action Repertoire

The *action repertoire*, first introduced in Section 7.3.2.2, provides much of the underlying functionality for the style axis model. In addition, it underpins the way in which the robot understands its own action capabilities in order to respond to initial requests for action from the human. Compare this kind of action selection with the process of attaining joint action attention by selecting a reference action during shaping. Once an action is present in the robot's scene sequence, there are many ways to refer to that action, because the set from which it must be isolated is constrained along several aspects (e.g. temporal, ordering, compositional). However to at first select actions to be added to the scene, the problem is much more difficult: how can the human teacher, armed with incomplete knowledge of the robot's internal representations and abilities, refer to a specific action among all that the robot might know how to perform?

In human-human interactions, the problem is similar if not worse — the set of actions any human knows how to do is essentially infinite and in any case unsearchable — yet people solve it expertly through a combination of the richness of descriptive language and the social skills of recognizing contextual constraints applying to the actions that might be relevant. The robot does not possess such skills to the same degree, but the process that has been implemented operates on

roughly the same lines. Requests for action are treated as queries, and the action repertoire acts as a motion database, constraining the search space according to the available query parameters and attempting to return an action that is at least close enough that it can be later shaped into one that is acceptable.

One of the most powerful query parameters has already been discussed in Section 7.3.2.2 — the symbolic class to which the action belongs — but this causes the Symbol Grounding Problem to rear its head again. The human currently does not have the ability to initially request actions by their class, because it cannot be assumed *a priori* that the human and the robot agree on the class terminology. The class can only be used for further queries once an action has been initially selected, in order to find other actions of the same class via style axis traversal.

Instead, the action repertoire implements query parameters to constrain the search that are considered basic enough to be acceptable as *a priori* shared symbolic knowledge. The most important of these is query by body region, which is acceptable solely because Leonardo is an anthropomorphic humanoid robot. Body region can be specified explicitly through verbal speech components — such as the words "head" or "arm" — or implicitly via Leonardo's inward attention mechanism, which can both isolate body regions by type and specify them within a type to disambiguate between multiple examples of a type (e.g. left vs right).

In order to implement this, actions that are loaded into the action repertoire can contain a tag that identifies which body regions the motion might apply to. Currently, this is specified by the designer when the action encapsulation is created. Ultimately, it would be preferable for the robot to learn these on its own, which could easily be accomplished using the other components that have been developed as part of this thesis. The robot would simply need to go through an extended Imagining phase at startup time, during which it played through all of its actions on its imaginary body model (Section 3.3) and concurrently monitored which areas of its internal attention system (Section 3.4) became activated, automatically applying these as tags to the action. This has not been implemented solely due to the desire not to extend the robot's already significantly tedious startup procedure during testing.

During the Shaping phase of the robot's scene, recall that actions could be referred to based on their cardinal ordering within the sequence. This is intuitive as well as possible by virtue of the inherent ordering provided by the unidirectional flow of time. The action repertoire has no such inherent ordering, but nevertheless it has been designed to allow queries using cardinal constraints based on the order actions are entered into the action repertoire by the designer.

There is little justification for allowing this — it is not a natural interaction, and it relies on information about the ordering of actions within the action repertoire that cannot be assumed *a priori* — other than the simple requirement for a mechanism of "last resort" for enabling the human to extract actions from the action repertoire. It is preferred that the teacher use body region constraints and transmodally bracketed demonstrations to extract an action of the correct class that can then be refined with interactive shaping. If all of these fail to ultimately specify a desired action, however, the human may at least proceed with the interaction by stepping through whatever potential actions remain after applying the aforementioned constraints to the action repertoire. A conceptual overview diagram of the action database functionality provided by the action repertoire is shown in Figure 7-26.

The other primary function of the action repertoire, besides action lookup, is the management of the difference between action prototypes and actual actions. Recall that actions in general

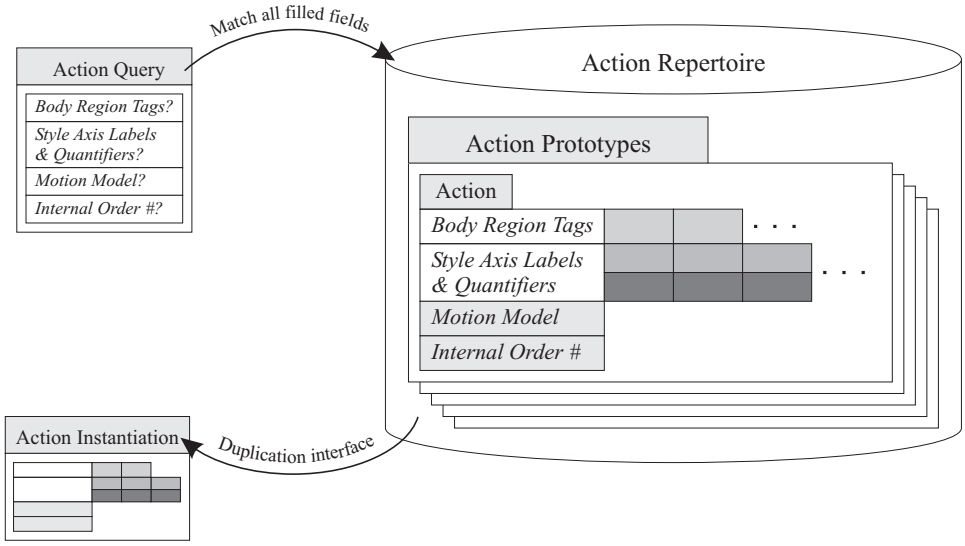


Figure 7-26: The action repertoire acts primarily as an action database. Action queries may contain any of the fields shown, and all fields present must be matched in order to return a result. Actions that are returned are duplicates of the prototypes in the repertoire, so that these instantiations may be safely modified elsewhere in the acting system.

encapsulate motions; motions are unique, whereas any number of actions may trigger a particular motion (Section 7.3.1). It is therefore not feasible for the action repertoire to contain every single action that the robot might perform after shaping; instead, it contains prototypes of each pre-shaped motion.

As introduced in Section 7.3.2.2, the nature of the shaping process requires that care be taken to distinguish action instantiations from action prototypes. This primarily concerns the persistence of qualities that have been applied via shaping. When a particular action in a scene is altered by shaping, it should remain as it was shaped whenever it is expressed in that scene. However, other instances of that action in different points in the scene, or in different scenes, should not be affected. The action repertoire takes care of this in the way that it returns the results of action queries.

Actions that are added to the action repertoire are qualitatively no different from other actions in the robot's action system — there is no particular execution detail that distinguishes action prototypes from action instantiations or prevents them from being used interchangeably to generate actual robot behavior — but in order to be added, they must implement a simple *duplication* interface. When the action repertoire returns an action in response to a query, it in fact returns an duplicate of the action that is identical other than its unique identifier within the action system. This action can then be shaped or deleted at will without affecting other instances of the same action.

Of course, enabling the actual shaping requires some lower level support, because many actions refer directly to the underlying unique motor primitives (e.g. the Speed Manager, in Section 7.3.2.2). In addition, there needs to be a way to compare action instantiations with action prototypes, without necessarily knowing if one is a prototype or not, in order to be able to determine whether two actions do qualitatively the same thing. This is due to the default treatment of actions in `C5M`. Actions are a subclass of the `NamedObject` class, meaning that the identifier that uniquely refers

to an action is its name, a text string. This requires that each instantiation of an action from a prototype must have a different name, and thus the name is not a reliable indicator of whether two actions actually perform the same activity. Instead, other attributes of the actions must be used for comparison. The action repertoire performs this functionality as well, comparing actions for equivalence based on the internal references it uses for searching: their class tags, style axis values, and body regions. Figure 7-27 illustrates the ways that the action repertoire is used by the overall acting system.

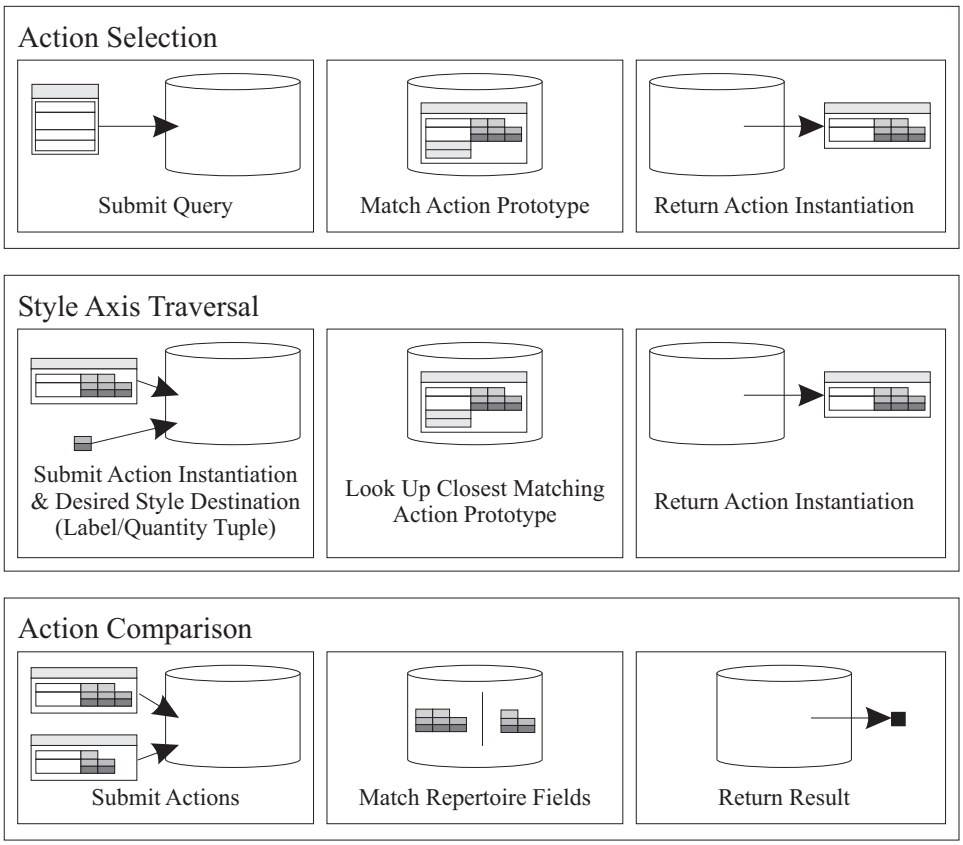


Figure 7-27: Overview of the ways in which the action repertoire is used by the acting system. The action repertoire performs a straight database lookup function for supporting action selection, a directed query match for style axis traversal, and a general comparison function for action matching independent of prototype/instantiation duality.

7.4.3.3 Other Sequenced Action Encapsulations

Several additions to the basic action functionality provided by the `C5M` action model (overviewed in Section 7.2.1) have already been described: the self-modeling of duration (Section 7.2.2.1), the overlay functionality (Section 7.3.2.2), and the duplication interface described above. Just a couple of other additions to the archetypal action remain to be summarized here.

First of these is a mechanism to facilitate negative and corrective feedback during the shaping process. Once again, interactive shaping is characterized as "learning by doing": the robot tests

and validates its progress by performing each refined action as it is instructed. Given the action model that has been described, this means that the action must have been actually modified in order for the robot to perform it. If the robot gets the modification wrong in some way, or if the human changes his or her mind about the modification in the first place, the robot needs a way to revert the sequence (and therefore the action) to its former state.

Since the action repertoire provides the functionality of generating new actions from prototypes, as well as qualitative comparison of basic actions, one way of accomplishing this could be to simply replace the incorrectly modified action with a new instantiation of the original prototype. However, this approach has a fundamental drawback in that it would result in the loss of any other shaping that had been performed on that action prior to the instance being canceled. This mechanism can be used as a last resort, but better options are available.

Instead, atomic actions suitable for shaping are encouraged to support a *backup* interface that allows the most recent shaping to be "undone" without the shaping system having to explicitly keep track of earlier states of the action. The individual actions (which are implemented as Java classes) are responsible for implementing this interface, since they are in the best position to determine which information in their container needs to be saved. This arrangement saves the shaping system from a lot of overhead, as the list of things that could potentially have been altered by the shaping process — ranging from motion animations triggered to direct joint parameters — is long, and forcing each action to make all of this information externally visible would be unwieldy. Applying feedback during shaping is thus mostly transparent from the point of view of the teaching system.

The other major encapsulation detail, that was alluded to in the introduction to Scheduled Action Groups (in Section 7.2.1), is the way that the timing network is broken up into subnetworks according to turn-taking behavior. In the context of dramatic acting, turn-taking corresponds to cueing. At present, the system is limited to verbal cues, which are converted into speech commands by the perceptual preprocessing system.

The segments of an actor's scene between cues are essentially arranged in a linear sequential order according to the script, so there is no need to have to deal with branching of the subnetwork chain. The implementation of cue segmentation is therefore quite straightforward. Each contiguous scene segment is represented by its own Scheduled Action Group, and these sequential subscenes are connected together into a linked list. When the acting playback system reaches the end of a Scheduled Action Group, it waits for a cue speech command to trigger the next one. This is illustrated in Figure 7-28.

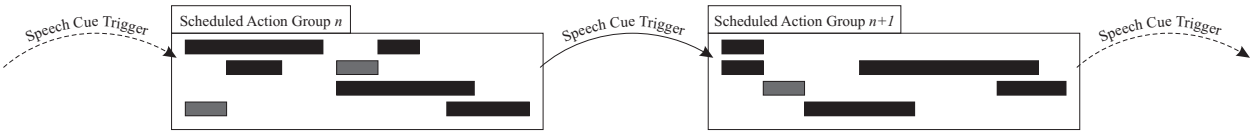


Figure 7-28: Use of speech cue triggers to chain together self-contained Scheduled Action Groups, allowing the robot to have portions of its acting sequence operate independent of internal timing.

The cue points themselves are set explicitly by the human during teaching, by using timing commands that inform the robot of the existence of a cue. For example, the natural language sentence that might be condensed into such a timing command might be "After you say your line, wait for my line". Receiving a cue timing command during shaping causes the shaping system

to begin a fresh Scheduled Action Group at the end of the list. Due to the network architecture underlying the Scheduled Action Groups, recombining them is straightforward if desired.

As mentioned in Section 7.2.2.2, there is a special case of spacer intervals that allows them to commence Scheduled Action Groups, in order to allow the robot to convey hesitancy or unresponsiveness in reaction to cues. The converse arrangement, in which the robot is so eager that it interrupts the human's cues, is more difficult to implement. Due to the limitations of the speech processing system that is used to detect cues, this behavior can not be instructed without some prior hacking of the speech grammar. If the cue line is modified to incorporate multiple cue tags, at different places in the line, the robot can be instructed to interrupt by telling it to wait for the appropriate cue in the resulting cue sequence.

7.4.4 Results and Evaluation

The work described in this chapter has achieved results in two main areas. First, it provides extensions to the `C5M` system that enable the autonomous production of significantly more complex interactive output. Second, it has demonstrated the application of these extensions with a novel interaction design that showed the potential of this approach for synthesizing robot communications through directed, real-time instruction.

In summarizing these results I will discuss these areas in order, first emphasizing what can be done with the expanded `C5M` functionality that was not previously possible. Primary among these is the ability to teach the robot complex sequences of rehearsed actions. The ethology-inspired architecture underlying `C5M` was originally most concerned with the connection of perception to action. However this left ambiguous the question of how to represent connections between actions that did not depend on perceptual sequencing inputs. The robot needed a way to execute sequences "with its eyes closed", as it were. A possible way to achieve this did exist earlier, by feeding internal action completion signals back through the perceptual system and using these to fire percepts that were set to trigger the next action in the sequence. Besides the obvious difficulties with unintentional sequence progression raised by this method, it does not account for sequences containing temporal relationships between actions other than *before* and *after*.

This leads directly to the next area of new functionality, the introduction of timing. Due to the immediate stimulus-response nature of purely percept-driven action, the ability to time the interrelation of actions was previously rudimentary, limited primarily to the `DO-UNTIL` conditions of action-tuples. The interval-based network architecture underlying the Scheduled Action Group provides support for all of the qualitative primitive temporal relations applied to actions. Furthermore, precise quantitative timing is also enabled through the use of spacer intervals. It is therefore now possible autonomously to schedule arbitrarily timed sequences, which previously could only have been done in advance by creating monolithic pre-timed animations or custom scripted action-tuples. In addition, the atomic actions used to assemble these sequences need not contain their own inherent timing information to be precisely scheduled, and they are able to be rearranged while preserving their local context with the hierarchical extensions to the interval network.

A further area of improved functionality is in the ability to influence the style of the robot's action output, beyond the application of timing to this end. Earlier incarnations of robots driven by `C5M` contained motor systems that were essentially an arbitrary collection of different ways to alter the characters' motion, at the designer's discretion. This might consist, for example,

of weighted blends, pre-designed verb-graph connectivity, and instantiations of my own direct joint motor system. Design of interactive methods for adjusting style under these conditions was challenging in its complexity. The placement of all of these beneath the umbrella of the linear Style Axis Model presents a more unified interface to the human interacting with the robot, and thus allows interactions to be designed to treat all style modifications in the same way (e.g. as though everything is represented by a weighted blend).

Although the added functionality has extended the architecture beyond the immediate connection of perception-action framework, architecturally all of this work has continued the theme of the unified `C5M` action model. As is the case with other action groups, the Scheduled Action Group itself subsumes the Action interface, allowing it to be inserted into action-tuples or other action groups as any ordinary action would be. Similarly, extensions to the atomic action definitions themselves allow action instantiations to modify themselves and maintain control of increased amounts of internal state. Self-timed actions monitor their own execution and keep estimates of their likely duration; overlay actions simplify whole-body control by providing control bridges across multiple motor systems; and the Speed Manager offers motion rate control at the action level, which was previously unavailable, via a transparent interface.

The increased power available to individual actions required better management of the duality between action prototypes and instantiations. The Action Repertoire contributed this functionality, enforcing reusability of actions and providing dependable adhesion of acquired characteristics, while also offering database-style lookup capabilities. With this construct it is now possible for the robot to perform a new kind of action-space discovery, cloning and modifying existing actions without fear that earlier learned sequences using those actions will be adversely affected. Moreover, it is now possible for the first time for the robot to traverse its action space according to the style and the body region involved.

Finally, this work reinforced the `C5M` action metaphor in areas where it had previously diverged, by encapsulating feedback-driven behaviors into higher level actions. Behaviors such as eye contact and lip synchronization used continuous firing of low-level actions, for example eye movements and eyelid blinks, to produce a qualitative overall motion output, but were not controllable at that qualitative level. It was thus unclear how these fit in to the generalized action model. Placing these behaviors within action structures inserted into the repertoire allowed them to be treated as single communicative entities to be holistically modified when desired as part of the interaction.

As a proof of concept of the generation of interactively designed communications, and to evaluate these computational additions, this work resulted in a novel demonstration interaction (Figure 7-29). In this demonstration, a human teacher was successfully able to multimodally transfer to the robot a communications sequence which was itself multimodal, and then shape and manipulate this sequence, all solely through a face-to-face tutelage process. To my knowledge this is the only system to have attempted this procedure with a robot to date.

The interaction involved a synthesis of modalities incorporating auditory speech commands, visually extracted physical demonstration, and tactile sensing of directed touch. All of these elements have been used in human-robot interaction before, but the utilization of all of them in a single interaction is less common. Moreover, the novelty of this interaction stems as much from what is absent as from the modes that are employed. The human teacher is able to coach the robot's performance without needing to write any script or code, and without needing to know many aspects of the inner workings of the robot that would ordinarily need to be known in or-

Figure 7-29: Still images from a demonstration with Leonardo of the interactive system for communications teaching and shaping.

der to write such scripts. Furthermore, no visualization tools are available to the teacher other than the performance and feedback behavior of the robot itself. In this sense, and by design, it is more similar to working with another human than to typical methods of interacting with computer systems.

The interaction further supported the establishment of common ground between the human and the robot in the presence of implicit prior language knowledge, in the form of the mechanism for grounding style symbols. This system of hypothesizing connections between internal adverbs and spoken labels was designed with theoretically optimal performance in terms of spanning the joint space, through incorporating the ability to simultaneously apply corrective feedback and generate additional hypothesis connections. Of course, this performance metric assumes that the human does not make any labeling mistakes. However, the hypothesis system also supports the correction of human-generated errors through its corrective feedback system.

An honest evaluation of this demonstration would not be complete without discussion of the aspects that leave something to be desired. Principal among these is the absence of any real dialog component. The auditory modality consists only of spoken commands directed at the robot by the human. However the robot is unable to respond in kind. When a human teacher directs a human learner, there is almost always the facility of bidirectional spoken communication to fall back upon when the interaction enters an ambiguous state. Although much can be inferred from body language and other contextual cues, there is no true substitute for natural language explication in complex situations. Although the demonstration interaction makes it possible to coach the robot through a wide range of performance dynamics, it is still possible to place it into a misunderstood state which it cannot resolve via its feedback output mechanisms, particularly in the presence of input errors from the speech processing subsystem. This is true of almost any open-ended system, but mimicking human interactivity with a robot presents particular challenges in this regard due to the expectations associated with the medium. It needs to at least be understood that the robot is making potentially erroneous inferences from contextual cues, and that higher than normal levels of performance monitoring, explicit descriptive statements, and restarting of tasks should be expected.

In the case of this interaction, the amount of context available for the robot to make inferences is limited. Much of the contextual information used to resolve action selection commands, for example, comes from temporal locality of reference. There are limits to the resolution of this information as an area of the robot's timeline becomes more densely populated or populated with more similar actions (e.g. actions with the same body region tags). Action selection commands theoretically could not be resolved even with explicit indexing statements if two actions involving the same body region are synchronized precisely together. However this can not currently occur due to limitations of the motor system that do not allow such actions to actually execute concurrently.

The hierarchical interval network structure also introduces context limitations when it comes to timing adjustment. This is primarily due to the way the command system is set up, involving only a single reference action per command. To understand how this limitation is manifested, consider an example timing adjustment in which an action is to be shifted later in respect to its parent but earlier in respect to its child. This modification will require two separate speech commands, because when the action is shifted later with respect to the parent, its child is also shifted along with it to preserve the existing temporal context. Therefore although the desired modification is ultimately possible, the two-stage method of achieving it may not be a strictly natural spoken interaction.

Finally, although the linear style axis model is a convenient construct from an interface stand-

point, there can be potential problems with applying it to certain underlying motor capacities — in particular, multidimensional blends that have nonlinear style manifolds. In other words, by making multiple traversals along different style axes, it may be possible to access regions of motor blend space that the designer may not have expected, including ones that could be potentially damaging to the robot. For example, if dependencies exist between facial DOFs that result in the presence of maximum joint differentials, it might be possible to exceed these with a blend between extremes of facial expressions that do not lie along the same style axis — say, for instance, extreme happy and extreme angry. In these cases not much can be said other than that it is up to the designer to test motor blends thoroughly and ensure that the adverb converters protect against accessing such undesirable areas of blend space.

Chapter 8

Conclusions

The work in this thesis spans a number of disparate areas of communication, both in terms of the communications methodologies employed and the underlying communications technologies involved. Moreover, it spans four very different sets of robotic hardware and software architectures. Nevertheless, all of this work is tied together by a focus on the fact that human-robot communication entails special aspects germane to the domain of a physical robot. It should not be assumed that we can necessarily simply apply existing HCI interface metaphors and communication schemes to interactions with a physical entity that communicates multimodally in manners that include those of living beings. Similarly, it also should not be assumed that familiar human-human interaction methods will necessarily just work, even if able to be implemented, without careful examination of the factors that differentiate the robotic compatriot from its human counterparts. In terms of adaptability to complex, nuanced communications, robots themselves are currently not algorithmically equipped to offer humans a great deal of latitude when it comes to interaction design. Coördinating human-robot communication is therefore essentially an engineering design problem that must be approached with care and predictive acumen.

In summary, then, this thesis is an engineering document that attempts to chronicle efforts to design communication skills for robots by bringing together inspiration from a broad literature of robotics, computer science, human factors engineering, animation, cognitive science and psychology. Through specific designs on specific robots, it attempts to provide an overview of the design of robots for human communication, the integration of communications mechanisms, and the development of useful infrastructure for these two broad aims. In concluding the thesis, I will revisit the claims of contributions made in Section 1.4, and summarize once more the justifications and evaluations of these ten claims.

The first contribution is the development of a vision-based "imitation game" for Leonardo that allowed the robot to generate a correspondence map between its body and that of a human teacher from a brief, real-time interaction. The naturalness of the imitative interaction absolved the human teacher from needing to know the details of the motion capture algorithms or calibration details, and the usability of the system was demonstrated by allowing the human to "puppeteer" the robot's body in real time, as it continuously mimicked the human's pose, directly following the interaction. This mapping was also successfully used to drive the robot's mental modeling of the human's body during subsequent communications interactions.

The novelty of this contribution stems from the informal, interactive nature of the method, requiring less specialized training, and from the emphasis on human judgement in the mapping process. In justification of the latter, the nature of the correspondence must be such that it preserves the communicative content of the physical activity, so the criteria in evaluating the correspondence of a particular body pose may vary for a given pair of skeletons depending on the pose. For example, for some poses, such as a fist held in the air at right angles to the elbow, the joint angles are important. For others, such as a hand held in front of the face, it is the spatial position of the hand that is important rather than the joint angles, which could be different for a body with different skeleton segment lengths. Fully automated techniques that must use purely mathematical metrics for correspondence would be inflexible in these cases, but keeping the human "in the loop" allows the teacher to use his or her contextual judgement to decide on the best match to ensure that the communicative intent of the pose carries through.

The second contribution is the design of a mental self-modeling framework for the Leonardo robot, in which it utilizes additional identical instantiations of its ordinary cognitive structures in order to make predictions and perform motion recognition on time series of arbitrary pre-chosen motion features. An important part of this contribution is the ability for the modeling components to convert stored and incoming sequences into different feature representations by simulating them on the robot's physical self-model. As a result of this architectural contribution, the robot was able to make comparisons of observed motions against its own motion repertoire according to multiple features, and thus correspondence metrics, and was able to accurately recompute its action scheduling based on simulation of the sequence execution.

While many architectures contain a self-modeling component, two aspects of this one are novel. Much of the self-modeling that is currently done concerns the use of a physical motor model that can be used to make physical predictions, and this is partly what is done here also. However, disagreement exists in the literature as to the level of descriptiveness of the representation that should be used for the model — whether a high fidelity representation that enables prediction for fine control tasks, or a generic representation that might more easily capture some general characteristics of an action, like the appearance of an aerobic exercise. So firstly, the system in this thesis is thus unusual because it uses a general intermediate representation that can be used, in conjunction with the model, to re-express a sequence in different motion representations depending on what information is most relevant at a particular instance.

Secondly, this system is distinct in that it reuses other cognitive structures besides the physical body model. Since the output of the robot is not determined just by the sequence of physical actions, but also by the decision mechanisms generating that sequence, reuse of the decision structures enables the simulation of actions within their local context. While this is primarily used for getting the timing correct when regenerating action sequence schedules, it has implications for action recognition as well. Though it may not result in direct improvements such as better recognition rates, it gives the designer the luxury of not being tied to task-dependent models and representation-dependent matching algorithms. The models are the same ones that the robot possesses for its normal activities, and the representation can be changed to suit a particular matching algorithm at will.

The third thesis contribution is the concept of the inward attention system and its implementation on Leonardo. Practically, this system allowed the robot to successfully interpret ambiguous verbal references on the part of the human teacher when they concerned the configuration of the

robot's physical form. More generally, it provides a framework for the robot to reason about the relevance of its own state to communications with a human partner; reasoning that could ultimately apply to situations other than ambiguities in direct instruction.

This is the essence of the importance of this concept. In order for introspection to be a useful counterpart to perception, there needs to be a regulatory mechanism that helps disambiguate and segment the most salient components from the robot's mental simulation apparatus much as conventional attention systems are employed to extract the most salient features from its perceptual apparatus. For the practical application of resolving symmetry of the self-model during instruction, for example, simple rules such as "the body part that moved last is the referent" could be — and are — used, just as simple perceptual rules are used as components of outward attention systems. The difference between an attention system and a loose collection of feature percepts is their organization into a cohesive whole that takes into account activation, habituation and interplay of features, as this does. To my knowledge this is the first such proposal of an inward attention system that applies these techniques directly to the robot's self-conceptualization.

The fourth contribution is a framework for minimal communication during large-scale interaction with a novel, non-anthropomorphic, lifelike robot. This framework was developed as an extension of a proposed definition for minimal HRI awareness, an area which to date has not received attention from the wider human-machine interaction community. The implemented perceptual system based on this framework drove a complex robotic installation for several days at a heavily attended major US conference, to generally positive responses. These impressions on the part of attendees were verified through a self-report questionnaire.

The principal novelty in this contribution, besides that of the robot and installation itself, is the focus on baseline communication skills — the enabling of communication in situations of minimal apparent common ground, such as direct physical correspondence, and the suggested rules for designing systems under these conditions. To date this topic has been neglected in the HRI community, though attention is beginning to turn towards this area, particularly in terms of impoverished physical communication for robots in heavily constrained task domains such as search and rescue, and I foresee it attracting further interest as specialized robots become generally more commonplace in society.

The fifth contribution is the concept and implementation of a multimodal gestural grammar for deictic reference in which gestures as well as speech fragments are treated as grammatical elements. I posit a number of novel benefits for this arrangement. Parsing speech and gesture together into queries for a spatial database component in what is essentially a message passing architecture suggests a highly modular implementation in which the algorithms for each module need only support a predefined communications interface in order to be able to be replaced and upgraded with relative ease. Recursion in the grammar allows any number of visually identical object referents to be simultaneously determined, even if their relative placement is within the margin of human pointing error. The spatial database approach also allows hierarchical constraints to be included in spatial queries, facilitating *pars-pro-toto* and *totum-pro-parte* deixis.

The implementation of the system was largely evaluated through its performance in the Robonaut Demo 2, a live demo for the fund manager of the DARPA MARS (Defense Advanced Research Projects Agency Mobile Autonomous Robot Systems) program. During this demo, which proceeded without error, the robot had to successfully recognize an ordered labeling of a collection of small, visually indistinguishable lug nuts on a wheel. Graphical data collected during the demo showed

that the pointing gestures would have resulted in mislabelings if the multiple simultaneous labeling feature of this system was not present. The modularity of the system was later demonstrated by replacement of all of the non-core components — vision, speech and the spatial database.

The sixth contribution is the formulation of the "behavioral overlay" model for enabling a behavior-based robot to communicate information through encoding within autonomous motion modulation of its existing behavioral schemas. In a behavior-based robotic system, schemas operate largely independent of centralized control, so a principal benefit of the overlay model is that it allows general motor communication design to be separated from schema design, reducing the complexity of individual schemas and the distributed design overhead. The applicability of this model is also independent of the particular instrumental functions and hierarchical arrangements of the schemas themselves.

The model as expressed in this thesis is therefore a general model, and was evaluated by an example implementation of the model on the humanoid robot QRIO for the purpose of autonomously producing body language communications. The demonstration of this implementation took three main parts. Equipped with an unmodified situated behavior schema, the behavioral overlay system was able to introduce a range of body language communication that could be autonomously varied based on the values of the robot's emotional system variables, as well as those of a long-term relationship system developed as part of this work. Furthermore, reflexive interactive behavior schemas, such as contingent head nodding during speech, could also be varied without modification of the schemas to incorporate explicit perception of the situational factors affecting them. Finally, additional body language schemas were developed, such as contingent torso leaning, demonstrating along with the addition of relationship variables that upgrades to the robot's schema repertoire can proceed without modification to the overlay system and vice versa.

The seventh contribution is the design of a comprehensive proxemic communication suite for the humanoid robot QRIO, and its implementation using behavioral overlays. Though the constraints of the collaboration agreement with Sony concerning the proprietary technology of the robot did not allow a formal study to be performed to verify the accuracy of the proxemic communications, the system was demonstrated live to high-ranking staff and I designed a prototype HRI user study in anticipation of the potential for conducting such an experiment in the future. The communications output during the live demonstration, which was able to be continuously varied based in part on the representations of individual relationships and the "personality" of the robot as mentioned above, was considered to be recognizable and appropriate by those familiar with the robot's capabilities.

The novelty of this contribution stems primarily from the fact that robot proxemics is largely uncharted territory, though it has begun to receive increasing attention in the time since this work was performed. The suite of proxemic behaviors that was selected was based on a careful review of the human proxemics literature, and is the most detailed implementation of proxemic communication on a robot to date. The addition of a representation for individual long-term relationships and the personality of individual robots was furthermore a novel inclusion in the QRIO software architecture.

The eighth contribution is the design of a representation for precisely timed robot action sequences based on modified interval algebra networks. The extensions to the interval algebra are the management of spacer intervals, which make possible precise relative timing with more flexibility during changes to the network than do unary node constraints, and the addition of interval hierarchies which allow context-dependent rearrangement of node clusters. The efficacy of this

arrangement was demonstrated by showing that the robot could convert a set of actions, each of which contained no inherent timing information relative to other actions, into a scheduled sequence that supported not only the relative interval relationships but more precise start point timing than the primitive relationships allow. Moreover, the network could be rearranged with more local context that the basic IA-network design, by preserving relationships between parent nodes and their children according to the additional stored hierarchy information.

The novelty of this contribution stems largely from the implementation within C5M, which previously had no mechanism for executing arbitrary timed sequences. Support structures that needed to be introduced to the architecture to support this were the interval algebra network computation functionality itself, the Self-Timed actions that kept track of their own durations and deactivation conditions, and the Scheduled Action Group encapsulation. The extensions made to the interval algebra are also new, perhaps not because they are particularly complex solutions but because once again the use of precise reschedulable action timing in human-robot interaction has not been comprehensively explored to date. In many non-communicative tasks only relative timing is important — for example, "add the coffee before the milk" — and precise timing is generated by direct feedback loops, as in catching a tracked ball, so precise pre-planned timing is rarely necessary. However as robots become better communicators and are endowed with the ability to use nuance, such subtle timing adjustments will prove increasingly necessary and therefore so will extensions to foundational representations such as these.

The ninth contribution is the design and implementation of the Style Axis Model for enabling a human director to request action-level stylistic modifications without having to know the details of motor-level representations or the robot's internal style labels. Prior to the development of this model, the only way to do linear style traversals within C5M was when the underlying motions had been set up as a weighted blend with an adverb converter. The effectiveness of this model was demonstrated by allowing logical style traversals to be applied among motions that had been designed by an animator as standalone animations rather than blends.

Although the separation of style from content is a popular topic in the field of computer animation, such systems typically make more information available to the animator, in the form of on-screen displays and interface controls such as timeline scrubbers. This system is novel in the sense that it centers around a direct spoken interaction much like a human teacher instructing another human. However most of the computational novelty of this system stems from its incorporation into the C5M system. Besides the addition of linear style traversals to unblended animations, several other interfaces have been added to C5M as part of this package. The Action Repertoire prototype database also allows changes, such as stylistic alterations, to be made to an action without affecting all other cases of that action within the behavior engine. The Speed Manager similarly allows speed changes to individual actions during run time, which is a previously unavailable feature. Lastly, the addition of Overlay Actions provides a new interface for coupling action instantiations across different motor systems.

The tenth and final contribution is the design and implementation in C5M of real-time, purely interactive multimodal communications sequence transfer from a human to a robot via a tutelage scenario. I believe this to be the first system in which a teacher can stand in front of a robot, coach it through a sequence with speech commands, physical demonstration and touch, and then adjust the sequence timing and the fashion in which the components are expressed, all without having to write any code or pre-arrange any script. Furthermore, the interaction functions in the absence of

any visualization for the teacher other than the output performance of the robot.

Once again, part of the novelty of this contribution stems partly from it treading new ground in terms of its design goals — the particular synthesis of perceptual and cognitive techniques into a unique tutelage scenario. A number of novel design and C5M components also underpin the contribution. In the former instance, the interaction made use of the style axis model via a mechanism for achieving common ground on stylistic terminology by connecting the human's style labels to the robot's style adverbs via a feedback-driven hypothesis system with theoretically optimal performance. Although in practice the system is limited by the need to explicitly design the human's style labels into the speech grammar, the future presence of speech recognition systems with larger vocabularies will make this kind of arrangement more attractive. In the latter case, the Direct Joint Motor System adds a new option for the control of character joints, the Backup and Duplicator action interfaces provide new methods for run-time action modification, and the encapsulations for dynamically generated behaviors such as lip synchronization and eye contact improve the applicability of these activities to open-ended interactions.

Overall, with these contributions I seek to demonstrate design choices and strategies for making robots capable of communicating more naturally with humans and facilitating more productive interactions between man and physically embodied machine. To that end I also hope to focus future embodiment towards conveying intuitive common ground, whether that be through anthropomorphism or other sound fundamental engagement. This work does not attempt to provide algorithmic panaceas; to the contrary, it shows that the most advanced existing methods must still be integrated and synthesized with great care and creativity. It is important that we not become wedded to particular much-touted technologies — these will improve, or pass, or certainly at least prove not to be sole solutions, as have artificial neural networks, Kalman filters, hidden Markov models... and so many others. As computer scientists and mathematicians we must understand them and know their uses and limitations, but as roboticists we must keep our eye fixed on the larger picture. This must be the design of robot architectures that are extensible and modular, and can make use of new and better underlying methods as they arrive. Moreover, these architectures must place a priority on ensuring that the synthesis of these methods is accessible to those who are not themselves roboticists. The better we coördinate our robots' communication skills, the more smoothly we can welcome them into the real world.

Appendix A

Acting Speech Grammar

```
<sizeModifier>          = ((much | (a (lot | bunch | heap))) {LARGE}) | (([just] a
                          (little | ([little] bit) | tad | smidgen | shade | fraction |
                          hair)) {SMALL});

<speedAdverb>           = ([<sizeModifier>] (((faster | quicker | ([more] (fast
                          | quick | quickly)) | (less (slow | slowly))) {INCREASE
                          SPEED}) | ((slower | ([more] slow | slowly) | (less (fast |
                          quick | quickly))) {DECREASE SPEED})));

<styleAdverb>           = ((<emotionAdverb> | <intensityAdverb> |
                          <generalStyleAdverb>) {STYLE});

<intensityAdverb>       = ([<sizeModifier>] (((([more] (intense | intensity |
                          energy | energetic | animated | enthusiasm | enthusiastic |
                          aggression | aggressive)) | (less (calm | subtle | demure)))
                          {INCREASE INTENSITY}) | (((less (intense | intensity |
                          energy | energetic | animated | enthusiasm | enthusiastic
                          | aggression | aggressive)) | ([more] (calm | subtle |
                          demure))) {DECREASE INTENSITY})));

<emotionAdverb>         = ([<sizeModifier>] ((([(more {INCREASE}) | (less
                          {DECREASE})] (((angry | anger) {ANGER}) | ((sad | sadness)
                          {SAD}) | ((happy | happiness) {HAPPY}) | ((calm | calmness |
                          neutral | peaceful) {NEUTRAL}) | ((surprise | surprised)
                          {SURPRISE}))) | ((angrier {INCREASE ANGER}) | (sadder
                          {INCREASE SAD}) | (happier {INCREASE HAPPY}) | (calmer
                          {INCREASE NEUTRAL}))) {EMOTION}));

<generalStyleAdverb>    = ([<sizeModifier>] ((([(more {INCREASE}) | (less
                          {DECREASE})] (((clumsy | clumsiness) {CLUMSY}) | ((precise
                          | precision) {PRECISE}) | ((fluid | fluidity) {FLUID}) |
                          ((jerky | jerkiness) {JERKY}) | ((flamboyant | flamboyantly
                          | expansive | expansively) {EXTROVERT}) | ((shy | shyly |
                          guarded | guardedly) {INTROVERT}))) | ((clumsier {INCREASE
                          CLUMSY}) | (jerkier {INCREASE JERKY})))));

<timingAdverb>          = ([<sizeModifier>] (((later {INCREASE}) | ((sooner |
                          earlier) {DECREASE})) {START-TIME}));
```

```
<timingSynch>              =  ((simultaneous | simultaneously | (at the same time) |
                              together) {SYNCHRONIZE START-TIME});

<timingImmediate>          =  ((now | immediately) {NOW START-TIME});

<timingAbsolute>           =  ([for] ((((((a | one) {ONE-SECOND}) | (((half a) | (a half))
                              {HALF-SECOND}) | (a quarter [of a] {QUARTER-SECOND}))
                              second) | (((two {TWO-SECOND}) | (three {THREE-SECOND})
                              | (four {FOUR-SECOND}) | (five {FIVE-SECOND}) | (six
                              {SIX-SECOND}) | (seven {SEVEN-SECOND}) | (eight
                              {EIGHT-SECOND}) | (nine {NINE-SECOND}) | (ten
                              {TEN-SECOND})) seconds))) {START-TIME});

<timingRelative>           =  ((for | until) (me | (my line) | (my cue)) {CUE
                              START-TIME});

<actionExplicit>           =  ((your | the) [<numberOrdinal>] ((arm {ARM}) | (face
                              {FACE}) | ((speech | line | lines) {SPEECH}) | ((torso
                              | body) {TORSO}) | (hand {HAND}) | (head {HEAD}) | (eye
                              {EYE})) [action | motion | movement] [<numberCardinal>]);

<actionImplicit>           =  (((that | the) (action | motion | movement)) | that | it);

<action>                   =  ((<actionExplicit> | <actionImplicit>) {ACTION});

<actionPluralExplicit>     =  ((<actionExplicit>) ([and <actionExplicit>]));

<actionPluralImplicit>     =  (((those) [actions | motions | movements]) | them);

<actionPlural>             =  ((<actionExplicit> {ACTION}) ([and <actionExplicit>
                              {ACTION}]));

<adverbalBasicCommand>     =  ([(do | make | (say {SPEECH})) <action>] (<speedAdverb> |
                              ([with] <styleAdverb> | <timingAdverb>));

<adverbalEmotionCommand>   =  ((((look | appear | be | act | (speak {SPEECH ACTION}))
                              <emotionAdverb>) | (make (a | your) <emotionAdverb> (face
                              | expression))) [<timingImmediate>]);

<adverbalTimingCommand>    =  (((do | (say {SPEECH})) (<action> | <actionPlural>))
                              (<timingImmediate> | (after [(another {INCREASE})]
                              <timingAbsolute>) | <timingSynch>)) | (((delay <action>)
                              | wait) (([(another {INCREASE})] <timingAbsolute>) |
                              <timingRelative>));

<adverbalCommand>          =  ([<robotname>] (<adverbalBasicCommand> |
                              <adverbalEmotionCommand> | <adverbalTimingCommand>));

<moreOrLess>               =  ((more {INCREASE}) | (less {DECREASE}));

<torsoCommand>             =  (((hold your body) | stand) [<moreOrLess>] (((erect
                              | upright | (up straight)) {TORSO ERECT}) | ((hunched
                              | slouched) {TORSO SLOUCH})))) | (lean [<moreOrLess>]
                              ((forward {TORSO FRONT}) | ((back | backward) {TORSO
                              BACK}))) | (turn [your body] [<moreOrLess>] ((left {TORSO
                              LEFT}) | (right {TORSO RIGHT})));

<armCommand>               =  ((hold | move) your arm {ARM}) ((out (ahead | front)
                              {FRONT}) | ([to your] ((left {LEFT}) | (right {RIGHT})))
                              | (like (this | that) {CONFIG})) | (from here {STARTPOINT});
```

```
<armCommandFinish>      =  ([[and] then [move [it]]] ((to here) | (like (this |
                           that))) {ARM ENDPOINT});

<headCommand>           =  (look [<moreOrLess> | (further {INCREASE})] {HEAD})
                           (([straight] ahead {FRONT}) | (left {LEFT}) | (right
                           {RIGHT}) | (((at me) | engaged) {EYECONTACT-ON}) | (((away
                           [from me]) | disengaged) {EYECONTACT-OFF}));

<eyeCommand>            =  ((make eye {EYE} contact {EYECONTACT-ON}) | (break eye
                           {EYE} contact {EYECONTACT-ON}) | (blink your eyes {EYE
                           BLINK}) | ([make your] eyes {EYE} [<moreOrLess>] ((open
                           {OPEN}) | (closed {CLOSE}))) | (((open {EYE OPEN}) | (close
                           {EYE CLOSE})) your eyes));

<handCommand>           =  (hold your [(left {LEFT}) | (right {RIGHT})] hand like
                           (this | that) {HAND CONFIG}) | (((do (this | that)) | (point
                           {POINT}) | (make a fist {FIST})) with your [(left {LEFT}) |
                           (right {RIGHT})] hand {HAND CONFIG});

<bodyCommand>           =  ([<robotname>] ((<torsoCommand> | <armCommand> |
                           <headCommand> | <eyeCommand> | <handCommand>) {BODY})
                           [<timingImmediate> | <timingSynch>]);

<readyQuestion>         =  (((ready [<robotname>]) | ([<robotname>] are you ready))
                           {READY QUESTION});

<yesNo>                 =  ((yes {YES}) | (no {NO}));

<feedbackResponse>      =  ([<yesNo>] [thats] (((good | better | right) {GOOD
                           FEEDBACK}) | ((bad | worse | (not right)) {BAD FEEDBACK})));

<styleCorrection>       =  ((no {NO}) thats <styleAdverb> {BAD FEEDBACK});

<styleOpposite>         =  (<styleAdverb> is ((the (opposite | reverse) of) |
                           ((opposite | opposed) to)) <styleAdverb> {OPPOSITE});

<finishSequence>        =  (((thats (all | (a wrap) | (it) | (your part))
                           [<robotname>]) | cut) {TASK DONE});

<restartSequence>       =  ((([(take it)] from the (top | beginning)) | ([lets]
                           (restart | replay))) {RESTART});

<stopGo>                =  (stop {STOP}) | ((go | action) {GO});

<continueSequence>      =  (((continue [from here]) | (keep going)) {CONTINUE});

<clearSequence>         =  (([[lets] start (over | again | ([over | again] from
                           scratch))) | (clear [your part]))) {CLEAR});

<repeatAction>          =  ([(show me [it | that]) | (do (it | that))] (again | (((one
                           more) | another) time)) {REPEAT});

<modeAssembly>          =  ([<robotname>] (((lets | (shall we)) learn) | (I will
                           teach you)) your part [<robotname>] {MODE ASSEMBLY});

<modeShaping>           =  ([<robotname>] (lets | (shall we)) work on your (part
                           | timing | act | acting | movement) [<robotname>] {MODE
                           SHAPING});

<modeReplay>            =  ([<robotname>] (([your (part | timing | act | acting |
                           movement)] looks [very | really] good) | (lets (try |
                           practice) (it | that) [out] [together])) [<robotname>]
                           {MODE REPLAY});
```

```
<generalActingCommand>      =  (<readyQuestion> | <yesNo> | <feedbackResponse> |
                               <styleCorrection> | <styleOpposite> | <finishSequence>
                               | <restartSequence> | <stopGo> | <continueSequence>
                               | <clearSequence> | <repeatAction> | <modeAssembly> |
                               <modeShaping> | <modeReplay>);

<numberOrdinal>             =  ((first {1}) | (second {2}) | (third {3}) | (fourth {4}) |
                               (fifth {5}) | (sixth {6}) | (seventh {7}) | (eighth {8}) |
                               (ninth {9}) | (tenth {10}));

<numberCardinal>            =  ((one {1}) | (two {2}) | (three {3}) | (four {4}) | (five
                               {5}) | (six {6}) | (seven {7}) | (eight {8}) | (nine {9}) |
                               (ten {10}));

<actingCommand>             =  (<adverbalCommand> | <bodyCommand> |
                               <generalActingCommand>);

<badSentence>               =  ((<sizeModifier> | <speedAdverb> | <intensityAdverb> |
                               <emotionAdverb> | <generalStyleAdverb> | <styleAdverb>
                               | <timingAdverb> | <timingSynch> | <timingImmediate> |
                               <timingAbsolute> | <timingRelative> | <actionExplicit>
                               | <actionImplicit> | <action> | <actionPluralExplicit> |
                               <actionPluralImplicit> | <actionPlural> | <moreOrLess> |
                               <numberOrdinal> | <numberCardinal>) {IMPROPER-PHRASE});

<robotname>                 =  (Leo | Leonardo);
```

Bibliography

[1] Adolphs, R., Tranel, D., and Damasio, A.R. The human amygdala in social judgment. *Nature*, 393(6684):470–474, 4 June 1998.

[2] Ali, K. and Arkin, R.C. Multiagent teleautonomous behavioral control. *Machine Intelligence and Robotic Control*, 1(2):3–10, 2000.

[3] Allen, J.F. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, 1983.

[4] Allen, J.F. Towards a general theory of action and time. *Artificial Intelligence*, 23(2):123–154, 1984.

[5] Allen, J.F. Time and time again: The many ways to represent time. *International Journal of Intelligent Systems*, 6(4):341–355, 1991.

[6] Allen, J.F. and Hayes, P.J. A common-sense theory of time. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence (IJCAI-85)*, pages 528–531, Los Angeles, California, 1985.

[7] Allen, J.F. and Hayes, P.J. Moments and points in an interval-based temporal logic. *Computational Intelligence*, 5(4):225–238, 1989.

[8] Althaus, P., Ishiguro, H., Kanda, T., Miyashita, T., and Christensen, H.I. Navigation for human-robot interaction tasks. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 2, pages 1894–1900, April 2004.

[9] Angelsmark, O. and Jonsson, P. Some observations on durations, scheduling and Allen's algebra. In Dechter, R., editor, *Proceedings of the 6th Conference on Constraint Programming (CP 2000)*, volume 1894 of *Lecture Notes in Computer Science*, pages 484–488. Springer-Verlag, 2000.

[10] Anger, F.D. and Rodriguez, R.V. Effective scheduling of tasks under weak temporal interval constraints. In Bouchon-Meunier, B., Yager, R.R., and Zadeh, L.A., editors, *Advances in Intelligent Computing - IPMU'94*, pages 584–594. Springer, Berlin, 1994.

[11] Aoyama, K. and Shimomura, H. Real world speech interaction with a humanoid robot on a layered robot behavior control architecture. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2005.

[12] Appenzeller, G., Lee, J.-H., and Hashimoto, H. Building topological maps by looking at people: An example of cooperation between intelligent spaces and robots. In *Proc. International Conference on Intelligent Robots and Systems*, Grenoble, France, November 1997.

[13] Assanie, M. Directable synthetic characters. In *Proceedings of the 2002 AAAI Spring Symposium on Artificial Intelligence and Interactive Entertainment*, 2002.

[14] Atkeson, C.G. and Schaal, S. Robot learning from demonstration. In *Proceedings of the 14th International Conference on Machine Learning*, pages 12–20. Morgan Kaufmann, 1997.

[15] Automatic Sync Technologies. ProductionSync. `http://www.automaticsync.com/lipsync/pdfs/AST_PSync.pdf`.

[16] Bailenson, J.N., Blascovich, J., Beall, A.C., and Loomis, J.M. Interpersonal distance in immersive virtual environments. *Personality and Social Psychology Bulletin*, 29:819–833, 2003.

[17] Bailey, D.R., Feldman, J.A., Narayanan, S., and Lakoff, G. Modeling embodied lexical development. In *Proceedings of the 19th Cognitive Science Society Conference*, pages 19–24, 1997.

[18] Bar-Shalom, Y. and Fortmann. *Tracking and Data Association*. Academic Press, San Diego, California, 1988.

[19] Barsalou, L.W., Niedenthal, P.M., Barbey, A.K., and Ruppert, J.A. Social embodiment. In Ross, B.H., editor, *The Psychology of Learning and Motivation*, volume 43, pages 43–92. Academic Press, 2003.

[20] Benford, S. and Fahlen, L. A spatial model of interaction in large virtual environments. In *Proceedings of the 3rd European Conference on Computer Supported Cooperative Work*, Milan, Italy, 1993.

[21] Beskow, J. and McGlashan, S. OLGA — a conversational agent with gestures. In *Proc. IJCAI-97 Workshop on Animated Interface Agents: Making Them Intelligent*, 1997.

[22] Bethel, C.L. and Murphy, R.R. Affective expression in appearance-constrained robots. In *Proceedings of the 2006 ACM Conference on Human-Robot Interaction (HRI 2006)*, pages 327–328, Salt Lake City, Utah, March 2006.

[23] Billard, A. and Dautenhahn, K. Grounding communication in autonomous robots: An experimental study. *Robotics and Autonomous Systems*, 24(1–2), 1998.

[24] Billard, A. and Dautenhahn, K. Experiments in learning by imitation — grounding and use of communication in robotic agents. *Adaptive Behavior*, 7(3/4):411–434, 1999.

[25] Blair, P. *Animation: Learning How to Draw Animated Cartoons*. Walter T. Foster Art Books, Laguna Beach, California, 1949.

[26] Blankenship, V., Hnat, S.M., Hess, T.G., and Brown, D.R. Reciprocal interaction and similarity of personality attributes. *Journal of Social and Personal Relationships*, 1:415–432, 2004.

[27] Bluethmann, W., Ambrose, R., Diftler, M., Huber, E., Fagg, A., Rosenstein, M., Platt, R., Grupen, R., Breazeal, C., Brooks, A.G., Lockerd, A., Peters, R.A. II, Jenkins, O.C., Matarić, M., and Bugajska, M. Building an autonomous humanoid tool user. In *Proc. IEEE-RAS/RSJ Int'l Conf. on Humanoid Robots (Humanoids '04)*, Los Angeles, California, November 2004.

[28] Blumberg, B. Action selection in Hamsterdam: Lessons from ethology. In *Proceedings of the Third International Conference on the Simulation of Adaptive Behavior*, pages 108–117, Brighton, England, August 1994.

[29] Blumberg, B., Downie, M., Ivanov, Y., Berlin, M., Johnson, M.P., and Tomlinson, W. Integrated learning for interactive synthetic characters. In *Proc. 29th Annual Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH '02)*, pages 417–426, San Antonio, Texas, 2002.

[30] Blumberg, B. and Galyean, T. Multi-level direction of autonomous creatures for real-time virtual environments. In *Proc. 22nd Annual Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH '95)*, pages 47–54, September 1995.

[31] Bolt, R.A. Put-That-There: Voice and gesture at the graphics interface. *ACM Computer Graphics*, 14(3):262–270, 1980.

[32] Bowlby, J. *Attachment & Loss Volume 1: Attachment*. Basic Books, 1969.

[33] Braitenberg, V. *Vehicles: Experiments In Synthetic Psychology*. MIT Press, Cambridge, Massachusetts, 1984.

[34] Brand, M. and Hertzmann, A. Style machines. In *Proc. 27th Annual Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH '00)*, New Orleans, Louisiana, 2000.

[35] Breazeal, C. A motivational system for regulating human-robot interaction. In *Proceedings of the 16th National Conference on Artificial Intelligence (AAAI 1998)*, pages 54–61, 1998.

[36] Breazeal, C. Proto-conversations with an anthropomorphic robot. In *9th IEEE International Workshop on Robot and Human Interactive Communication (Ro-Man2000)*, pages 328–333, Osaka, Japan, 2000.

[37] Breazeal, C. *Sociable Machines: Expressive Social Exchange Between Humans and Robots*. PhD thesis, Massachusetts Institute of Technology, 2000.

[38] Breazeal, C. *Designing Sociable Robots*. MIT Press, Cambridge, Massachusetts, 2002.

[39] Breazeal, C. Emotive qualities in lip-synchronized robot speech. *Advanced Robotics*, 17(2):97–113, 2003.

[40] Breazeal, C. Towards sociable robots. *Robotics and Autonomous Systems*, 42(3–4):167–175, 2003.

[41] Breazeal, C. Social interactions in HRI: The robot view. *IEEE Transactions on Systems, Man and Cybernetics—Part C*, 34(2):181–186, May 2004.

[42] Breazeal, C., Brooks, A.G., Gray, J., Hancher, M., McBean, J., Stiehl, W.D., and Strickon, J. Interactive robot theatre. In *Proc. International Conference on Intelligent Robots and Systems*, Las Vegas, Nevada, October 2003.

[43] Breazeal, C., Brooks, A.G., Gray, J., Hoffman, G., Kidd, C., Lee, H., Lieberman, J., Lockerd, A., and Chilongo, D. Tutelage and collaboration for humanoid robots. *International Journal of Humanoid Robots*, 1(2):315–348, 2004.

[44] Breazeal, C., Buchsbaum, D., Gray, J., Gatenby, D., and Blumberg, B. Learning from and about others: Towards using imitation to bootstrap the social understanding of others by robots. *Artificial Life*, 11(1-2), 2005.

[45] Breazeal, C., Kidd, C.D., Lockerd Thomaz, A., Hoffman, G., and Berlin, M. Effects of nonverbal communication on efficiency and robustness in human-robot teamwork. In *Proc. International Conference on Intelligent Robots and Systems*, 2004.

[46] Breazeal, C. and Scassellati, B. A context-dependent attention system for a social robot. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI 1999)*, pages 1146–1151, 1999.

[47] Breazeal, C. and Scassellati, B. Robots that imitate humans. *Trends In Cognitive Sciences*, 6:481–487, 2002.

[48] Brooks, A.G. Mechanisms and processing techniques for real-time active vision. Master's thesis, The Australian National University, Canberra, Australia, 2000.

[49] Brooks, A.G. and Arkin, R.C. Behavioral overlays for non-verbal communication expression on a humanoid robot. *Autonomous Robots*, 2006.

[50] Brooks, A.G., Berlin, M., Gray, J., and Breazeal, C. Untethered robotic play for repetitive physical tasks. In *Proc. ACM International Conference on Advances in Computer Entertainment*, Valencia, Spain, June 2005.

[51] Brooks, A.G. and Breazeal, C. Working with robots and objects: Revisiting deictic reference for achieving spatial common ground. In *Proceedings of the 2006 ACM Conference on Human-Robot Interaction (HRI 2006)*, pages 297–304, Salt Lake City, Utah, March 2006.

[52] Brooks, A.G., Gray, J., Hoffman, G., Lockerd, A., Lee, H., and Breazeal, C. Robot's play: Interactive games with sociable machines. In *Proc. ACM International Conference on Advances in Computer Entertainment*, Singapore, June 2004.

[53] Bruce, A., Knight, J., Listopad, S., Magerko, B., and Nourbakhsh, I.R. Robot improv: Using drama to create believable agents. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 4002–4008, San Francisco, California, April 2000.

[54] Bruce, A., Nourbakhsh, I., and Simmons, R. The role of expressiveness and attention in human-robot interaction. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 4, pages 4138–4142, 2002.

[55] Burke, J.L., Murphy, R.R., Coovert, M.D., and Riddle, D.L. Moonlight in Miami: A field study of human-robot interaction in the context of an urban search and rescue disaster response training exercise. *Human-Computer Interaction*, 19:85–116, 2004.

[56] Byrne, D. and Griffit, W. Similarity and awareness of similarity of personality characteristic determinants of attraction. *Journal of Experimental Research in Personality*, 3:179–186, 1969.

[57] Call, J. and Carpenter, M. Three sources of information in social learning. In Dautenhahn, K. and Nehaniv, C.L., editors, *Imitation in Animals and Artifacts*, pages 211–228. MIT Press, Cambridge, Massachusetts, 2002.

[58] Cassell, J., Bickmore, T., Campbell, L., Vilhjalmsson, H., and Tan, H. Human conversation as a system framework: Designing embodied conversational agents. In Cassell, J., Sullivan, J., Prevost, S., and Churchill, E., editors, *Embodied Conversational Agents*. MIT Press, Cambridge, Massachusetts, 2000.

[59] Cassell, J. and Vilhjalmsson, H. Fully embodied conversational avatars: Making communicative behaviors autonomous. *Autonomous Agents and Multiagent Systems*, 2:45–64, 1999.

[60] Cheng, G. and Zelinsky, A. Supervised autonomy: A framework for human-robot systems development. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, volume 2, pages 971–975, Tokyo, Japan, October 1999.

[61] Chi, D., Costa, M., Zhao, L., and Badler, N. The EMOTE model for effort and shape. In *Proc. 27th Annual Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH '00)*, pages 173–182, 2000.

[62] Christensen, H.I. and Pacchierotti, E. Embodied social interaction for robots. In Dautenhahn, K., editor, *Proceedings of the 2005 Convention of the Society for the Study of Artificial Intelligence and Simulation of Behaviour (AISB-05)*, pages 40–45, Hertfordshire, England, April 2005.

[63] Clark, H.H. *Arenas of Language Use*. University of Chicago Press, Chicago, 1992.

[64] Clark, H.H. and Brennan, S.E. Grounding in communication. In Resnick, B., Levine, J., and Teasley, S.D., editors, *Perspectives on Socially Shared Cognition*, pages 127–149. APA Books, Washington, DC, 1991.

[65] Clark, H.H. and Marshall, C.R. Definite reference and mutual knowledge. In Joshi, A.K., Webber, B.L., and Sag, I.A., editors, *Elements of Discourse Understanding*. Cambridge University Press, Cambridge, 1981.

[66] Clark, H.H., Schreuder, R., and Buttrick, S. Common ground and the understanding of demonstrative reference. *Journal of Verbal Learning and Verbal Behavior*, 22:245–258, 1983.

[67] Clouse, J.A. and Utgoff, P.E. A teaching method for reinforcement learning. In *Proceedings of the Ninth Conference on Machine Learning*, Los Altos, California, 1992. Morgan Kaufmann.

[68] CMU Sphinx Group. Open Source Speech Recognition Engines. `http://cmusphinx.sourceforge.net/`.

[69] Cohen, M.F. Interactive spacetime control for animation. *Computer Graphics*, 26(2):293–302, July 1992.

[70] Cohen, R. *Theater, Brief Version*. McGraw-Hill, Boston, 2003.

[71] Crandall, J.W., Goodrich, M.A., Olsen, D.R., Jr., and Nielsen, C.W. Validating human-robot interaction schemes in multitasking environments. *IEEE Transactions on Systems, Man and Cybernetics—Part A: Systems and Humans*, 35(4):438–449, July 2005.

[72] Crick, F. and Koch, C. The unconscious homunculus. In Metzinger, T., editor, *The Neuronal Correlates of Consciouness*, pages 103–110. MIT Press, Cambridge, Massachusetts, 2000.

[73] Darrell, T., Demirdjian, D., Checka, N., and Felzenszwalb, P. Plan-view trajectory estimation with dense stereo background models. In *Proc. International Conference on Computer Vision*, volume 2, pages 628–635, Vancouver, Canada, July 2001.

[74] Darrell, T., Gordon, G., Woodfill, J., and Harville, M. Integrated person tracking using stereo, color, and pattern detection. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Santa Barbara, CA, June 1998.

[75] Davies, M. and Stone, T. *Mental Simulation*. Blackwell Publishers, Oxford, UK, 1995.

[76] Davis, M. The role of the amygdala in fear and anxiety. *Annual Review of Neuroscience*, 15:353–375, 1992.

[77] de Vega, M. and Rodrigo, M.J. Updating spatial layouts mediated by pointing and labelling under physical and imaginary rotation. *European Journal of Cognitive Psychology*, 13(3):369–393, July 2001.

[78] Dean, T. and McDermott, D. Temporal data base management. *Artificial Intelligence*, 32:1–55, 1987.

[79] Dechter, R. From local to global consistency. *Artificial Intelligence*, 55(1):87–107, 1992.

[80] Dechter, R. and Pearl, J. Directed constraint networks: A relational framework for causal modeling. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI-91)*, pages 1164–1170, Sydney, Australia, 1991.

[81] Demirdjian, D., Ko, T., and Darrell, T. Constraining human body tracking. In *Proc. International Conference on Computer Vision*, Nice, France, October 2003.

[82] Demiris, Y. and Hayes, G. Imitation as a dual-route process featuring predictive and learning components: A biologically-plausible computational model. In Dautenhahn, K. and Nehaniv, C.L., editors, *Imitation in Animals and Artifacts*. MIT Press, Cambridge, Massachusetts, 2002.

[83] Dennett, D. *The Intentional Stance*. MIT Press, Cambridge, Massachusetts, 1987.

[84] Diderot, D. The paradox of acting. In Archer, W., editor, *Masks or Faces*. Hill and Wang, New York, 1957. Originally published 1830.

[85] DiSalvo, C., Gemperle, F., Forlizzi, J., and Kiesler, S. All robots are not equal: Design and the perception of humanness in robot heads. In *Proc. Conf. Designing Interactive Sys.*, pages 321–326, 2002.

[86] Dittmann, A. The role of body movement in communication. In *Nonverbal Behavior and Communication*. Lawrence Erlbaum Associates, Hillsdale, 1978.

[87] Dorigo, M. and Colombetti, M. *Robot Shaping: An Experiment in Behavior Engineering*. MIT Press, Cambridge, Massachusetts, 1997.

[88] Dorn, J. Temporal reasoning in sequence graphs. In *Proceedings of the 10th National Conference on Artificial Intelligence (AAAI'92)*, pages 735–740, San Jose, California, 1992. Morgan Kaufmann.

[89] Dourish, P. and Bellotti, V. Awareness and coordination in shared workspaces. In *Proceedings of the 4th Conference on Computer Supported Cooperative Work (CSCW '92)*, Toronto, Canada, 1992.

[90] Downie, M. behavior, animation, music: the music and movement of synthetic characters. Master's thesis, MIT Media Laboratory, Cambridge, Massachusetts, 2001.

[91] Drury, J.L., Scholtz, J., and Yanco, H.A. Awareness in human-robot interactions. In *Proceedings of the IEEE Conference on Systems, Man and Cybernetics*, Washington, DC, October 2003.

[92] Drury, J.L. and Williams, M.G. A framework for role-based specification and evaluation of awareness support in synchronous collaborative applications. In *Proceedings of the IEEE Workshops on Enabling Technologies Infrastructure for Collaborative Enterprises (WETICE)*, Pittsburgh, Pennsylvania, 2002.

[93] Dunbar, I. *Dog Behavior*. TFH Publications Inc., Neptune, New Jersey, 1979.

[94] Ekman, P. and Davidson, R.J. *The Nature of Emotion*. Oxford University Press, 1994.

[95] Endsley, M.R. Design and evaluation for situation awareness enhancement. In *Proceedings of the Human Factors Society 32nd Annual Meeting*, pages 97–101, Santa Monica, California, 1988.

[96] Eveland, C., Konolige, K., and Bolles, R.C. Background modeling for segmentation of video-rate stereo sequences. In *Proc. Computer Vision and Pattern Recognition*, pages 266–272, 1998.

[97] Fikes, R.E. and Nilsson, N.J. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2:189–205, 1971.

[98] Fincannon, T., Barnes, L.E., Murphy, R.R., and Riddle, D.L. Evidence of the need for social intelligence in rescue robots. In *Proc. International Conference on Intelligent Robots and Systems*, volume 2, pages 1089–1095, 2004.

[99] Fong, T., Nourbakhsh, I., and Dautenhahn, K. A survey of socially interactive robots. *Robotics and Autonomous Systems*, 42:143–166, 2003.

[100] Fridlund, A. *Human Facial Expression: An Evolutionary View*. Academic Press, San Diego, CA, 1994.

[101] Fujita, M., Kuroki, Y., Ishida, T., and Doi, T. Autonomous behavior control architecture of entertainment humanoid robot SDR-4X. In *Proc. International Conference on Intelligent Robots and Systems*, pages 960–967, Las Vegas, NV, 2003.

[102] Furui, S. Robust methods in automatic speech recognition and understanding. In *Proceedings of the 8th European Conference on Speech Communication and Technology (EUROSPEECH-03)*, pages 1993–1998, Geneva, Switzerland, 2003.

[103] Galef, B. Imitation in animals: History, definition, and interpretation of data from the psychological laboratory. In *Social Learning: Psychological and Biological Perspectives*. Lawrence Erlbaum Associates, 1988.

[104] Garvey, C. Some properties of social play. *Merrill-Palmer Quarterly*, 20:163–180, 1974.

[105] Giese, M.A. and Poggio, T. Morphable models for the analysis and synthesis of complex motion pattern. *International Journal of Computer Vision*, 38(1):59–73, 2000.

[106] Girard, M. Interactive design of 3D computer-animated legged animal motion. *IEEE Computer Graphics and Applications*, 7(6):39–51, June 1987.

[107] Glass, J. and Weinstein, E. SPEECHBUILDER: Facilitating spoken dialogue system development. In *Proceedings of the 7th European Conference on Speech Communication and Technology (EUROSPEECH-01)*, Aalborg, Denmark, September 2001.

[108] Gleicher, M. and Ferrier, N. Evaluating video-based motion capture. In *Proceedings of Computer Animation 2002*, pages 75–80, Geneva, Switzerland, 19–21 June 2002.

[109] Gockley, R., Bruce, A., Forlizzi, J., Michalowski, M., Mundell, A., Rosenthal, S., Sellner, B., Simmons, R., Snipes, K., Schultz, A.C., and Wang, J. Designing robots for long-term social interaction. In *Proc. International Conference on Intelligent Robots and Systems*, pages 1338–1343, 2–6 August 2005.

[110] Goetz, J., Kiesler, S., and Powers, A. Matching robot appearance and behavior to tasks to improve human-robot cooperation. In *Proceedings of the Twelfth IEEE Workshop on Robot and Human Interactive Communication (Ro-Man'03)*, pages 55–60, Millbrae, California, 31 October–2 November 2003.

[111] Gordon, R. Folk psychology as simulation. *Mind and Language*, 1:158–171, 1986.

[112] Gray, J. Goal and action inference for helpful robots using self as simulator. Master's thesis, MIT Media Laboratory, Cambridge, Massachusetts, 2004.

[113] Gray, J., Breazeal, C., Berlin, M., Brooks, A.G., and Lieberman, J. Action parsing and goal inference using self as simulator. In *Proceedings of the Fourteenth IEEE Workshop on Robot and Human Interactive Communication (Ro-Man'05)*, pages 202–209, Nashville, Tennessee, August 2005.

[114] Greif, L. *Computer-Supported Cooperative Work: A Book of Readings*. Morgan Kaufmann, 1988.

[115] Gullberg, M. Gestures in spatial descriptions. In *Working Papers 47*, pages 87–97. Lund University, Department of Linguistics, 1999.

[116] Gutwin, C., Greenberg, S., and Roseman, M. *Workspace Awareness in Real-Time Distributed Groupware: Framework, Widgets and Evaluation*. University of Calgary, 1996.

[117] Gutwin, C., Greenberg, S., and Roseman, M. Workspace awareness support with radar views. In *Proceedings of the 1996 Conference on Human Factors in Computing Systems (CHI '96)*, Vancouver, British Columbia, Canada, April 1996.

[118] Guye-Vuilleme, A., Capin, T.K., Pandzic, I.S., Thalmann, N.M., and Thalmann, D. Nonverbal communication interface for collaborative virtual environments. In *Proc. Collaborative Virtual Environments (CVE 98)*, pages 105–112, June 1998.

[119] Habili, N., Lim, C.-C., and Moini, A. Hand and face segmentation using motion and color cues in digital image sequences. In *Proc. IEEE Int. Conf. on Multimedia and Expo*, 2001.

[120] Hall, E.T. *The Hidden Dimension*. Doubleday, Garden City, NY, 1966.

[121] Hanafiah, Z.M., Yamazaki, C., Nakamura, A., and Kuno, Y. Human-robot speech interface understanding inexplicit utterances using vision. In *Late Breaking Results of the 2004 Conference on Human Factors in Computing Systems (CHI'04)*, pages 1321–1324. ACM Press, April 24–29 2004.

[122] Hara, F., Akazawa, H., and Kobayashi, H. Realistic facial expressions by SMA driven face robot. In *Proceedings of IEEE International Workshop on Robot and Human Interactive Communication*, pages 504–510, September 2001.

[123] Haritaoglu, I., Harwood, D., and Davis, L.S. W4: Real-time surveillance of people and their activities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):809–830, August 2000.

[124] Harnad, S. The symbol grounding problem. *Physica*, D:335–346, 1990.

[125] Hayes-Roth, B., Sincoff, E., Brownston, L., Huard, R., and Lent, B. Directed improvisation. Technical Report KSL-94-61, Knowledge Systems Laboratory, Stanford University, Palo Alto, California, September 1994.

[126] Heal, J. *Understanding Other Minds from the Inside*, pages 28–44. Cambridge University Press, Cambridge, UK, 2003.

[127] Hendrix, G.G. Modeling simultaneous actions and continuous processes. *Artificial Intelligence*, 4(3):145–180, 1973.

[128] Hoffman, G. and Breazeal, C. What lies ahead? Expectation management in human-robot collaboration. In Fong, T., editor, *To Boldly Go Where No Human-Robot Team Has Gone Before: Papers from the AAAI 2006 Spring Symposium*, pages 1–7, Menlo Park, California, 2006. American Association for Artificial Intelligence. Technical Report SS-06-07.

[129] Hoshino, K. and Kawabuchi, I. A humanoid robotic hand performing the sign language motions. In *Proceedings of the 2003 International Symposium on Micromechatronics and Human Science (MHS 2003)*, pages 89–94, 19–22 October 2003.

[130] Huber, E. and Baker, K. Using a hybrid of silhouette and range templates for real-time pose estimation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1652–1657, New Orleans, Louisiana, 2004. IEEE.

[131] Huls, C., Bos, E., and Claassen, W. Automatic referent resolution of deictic and anaphoric expressions. *Computational Linguistics*, 21(1):59–79, 1995.

[132] Iacoboni, M., Woods, R.P., Brass, M., Bekkering, H., Mazziotta, J.C., and Rizzolatti, G. Cortical mechanisms of human imitation. *Science*, 286:2526–2528, 1999.

[133] Institute for Applied Autonomy. Pamphleteer: A propaganda robot for cultural resistance. `http://www.appliedautonomy.com/pamphleteer.html`, 1999.

[134] Isard, M. and Blake, A. Condensation – conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5–28, 1998.

[135] Isla, D.A. The virtual hippocampus: Spatial common sense for synthetic creatures. Master's thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts, 2001.

[136] Isla, D.A., Burke, R., Downie, M., and Blumberg, B. A layered brain architecture for synthetic characters. In *Proc. International Joint Conference on Artificial Intelligence*, 2001.

[137] Ivanov, Y.A., Bobick, A.F., and Liu, J. Fast lighting independent background subtraction. *International Journal of Computer Vision*, 37(2):199–207, June 2000.

[138] Jazwinski, A.H. *Stochastic Processes and Filtering Theory*. Academic Press, New York, 1970.

[139] Jenkins, O.C. and Matarić, M. Automated derivation of behavior vocabularies for autonomous humanoid motion. In *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-03)*, Melbourne, Australia, July 2003.

[140] Jordan, P.W. *Designing Pleasurable Products: an Introduction to the New Human Factors*. Taylor & Francis Books Ltd., UK, 2000.

[141] Kaelbling, L.P., Littman, M.L., and Moore, A.W. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285, 1996.

[142] Kahn, K.M. and Gorry, A.G. Mechanizing temporal knowledge. *Artificial Intelligence*, 9(2):87–108, 1977.

[143] Kalman, R.E. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82:35–45, March 1960.

[144] Kanda, T., Hirano, T., Eaton, D., and Ishiguro, H. Interactive robots as social partners and peer tutors for children. *Human-Computer Interaction*, 19:61–84, 2004.

[145] Kaplan, F., Oudeyer, P.-Y., Kubinyi, E., and Miklosi, A. Taming robots with clicker training: A solution for teaching complex behaviors. In Quoy, M., Gaussier, P., and Wyatt, J.L., editors, *Proceedings of the 9th European Workshop on Learning Robots*, Lecture Notes in Artificial Intelligence. Springer, 2001.

[146] Kaur, M., Tremaine, M., Huang, N., Wilder, J., and Gacovski, Z. Where is 'it'? Event synchronization in gaze-speech input systems. In *Proc. 5th International Conference on Multimodal Interfaces (ICMI'03)*, pages 151–158, November 2003.

[147] Kendon, A. Current issues in the study of gesture. In Nespoulous, J.-L., Perron, P., and Lecours, A.R., editors, *The Biological Foundations of Gestures*, pages 23–47. Lawrence Erlbaum Associates, Hillsdale, NJ, 1986.

[148] Khadhouri, B. and Demiris, Y. Hierarchical, attentive, multiple models for execution and recognition (HAMMER). In Billard, A. and Dillman, R., editors, *Proceedings of the Workshop on Social Mechanisms of Robot Programming by Demonstration, International Conference on Robotics and Automation (ICRA-05)*, Barcelona, Spain, April 18 2005.

[149] Kiesler, D.J. The 1982 interpersonal circle: A taxonomy for complementarity in human transactions. *Psychological Review*, 90:185–214, 1983.

[150] Kiesler, S. Fostering common ground in human-robot interaction. In *Proceedings of the Fourteenth IEEE Workshop on Robot and Human Interactive Communication (Ro-Man'05)*, pages 158–163, Nashville, Tennessee, August 2005.

[151] King, S. and Weiman, C. HelpMate autonomous mobile robot navigation system. In *Proceedings of the SPIE Conference on Mobile Robots*, pages 190–198, Boston, MA, November 1990.

[152] Kitano, H., Tambe, M., Stone, P., Veloso, M., Coradeschi, S., Osawa, E., Matsubara, H., Noda, I., and Asada, M. The RoboCup 1997 synthetic agents challenge. In *Proceedings of the First International Workshop on RoboCup, International Joint Conference on Artificial Intelligence (IJCAI-97)*, Nagoya, Japan, 1997.

[153] Knapp, M. *Nonverbal Communication in Human Interaction.* Reinhart and Winston, Inc., New York, 1972.

[154] Kobsa, A., Allgayer, J., Reddig, C., Reithinger. N., Schmauks, D., Harbusch, K., and Wahlster, W. Combining deictic gestures and natural language for referent identification. In *Proc. 11th Conference on Computational Linguistics*, pages 356–361, Bonn, Germany, 1986.

[155] Koomen, J.A.G.M. Localizing temporal constraint propagation. In Brachman, R.J., Levesque, H.J., and Reiter, R., editors, *Proceedings of the 1st International Conference on Principles of Knowledge Representation and Reasoning (KR'89)*, pages 198–202, Toronto, Canada, May 15–18 1989. Morgan Kaufmann.

[156] Koons, D.B., Sparrell, C.J., and Thorisson, K.R. Integrating simultaneous input from speech, gaze, and hand gestures. In Maybury, M., editor, *Intelligent Multimedia Interfaces*, pages 257–276. MIT Press, Menlo Park, CA, 1993.

[157] Kopp, S. and Wachsmuth, I. A knowledge-based approach for lifelike gesture animation. In *Proc. ECAI-2000*, 2000.

[158] Kopp, S. and Wachsmuth, I. Model-based animation of coverbal gesture. In *Proc. Computer Animation*, 2002.

[159] Koster, B.E., Rodman, R.D., and Bitzer, D. Automated lip-sync: Direct translation of speech-sound to mouth-shape. In *Proceedings of the 28th Annual Asilomar Conference on Signals, Systems and Computers*, volume 1, pages 583–586, Pacific Grove, California, 1994.

[160] Kumar, V. Algorithms for constraint satisfaction problems: A survey. *AI Magazine*, 13:32–44, 1992.

[161] Kuniyoshi, Y., Inaba, M., and Inoue, H. Learning by watching: Extracting reusable task knowledge from visual observations of human performance. *IEEE Transactions on Robotics and Automation*, 10:799–822, November 1994.

[162] Kuniyoshi, Y. and Inoue, H. Qualitative recognition of ongoing human action sequences. In *Proc. International Joint Conference on Artificial Intelligence*, pages 1600–1609, 1993.

[163] Laird, J.E., Assanie, M., Bachelor, B., Benninghoff, N., Enam, S., Jones, B., Kerfoot, A., Lauver, C., Magerko, B., Sheiman, J., Stokes, D., and Wallace, S. A test bed for developing intelligent synthetic characters. In *Proceedings of the 2002 AAAI Spring Symposium on Artificial Intelligence and Interactive Entertainment*, 2002.

[164] Laszlo, J., van de Panne, M., and Fiume, E. Interactive control for physically-based animation. In *Proc. 27th Annual Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH '00)*, pages 201–208, New Orleans, Louisiana, 2000.

[165] Lathan, C.E. and Malley, S. Development of a new robotic interface for telerehabilitation. In *Workshop on Universal Accessability of Ubiquitous Computing (WUAUC '01)*, pages 80–83, Alcacer do Sal, Portugal, May 2001. ACM.

[166] Latoschik, M.E. and Wachsmuth, I. Exploiting distant pointing gestures for object selection in a virtual environment. In Wachsmuth, I. and Fröhlich, M., editors, *Gesture and Sign Language in Human-Computer Interaction*, volume 1371 of *Lecture Notes in Artificial Intelligence*, pages 185–196. Springer-Verlag, 1998.

[167] Lau, I.Y-M., Chiu, C-Y., and Hong, Y-Y. I know what you know: Assumptions about others' knowledge and their effects on message construction. *Social Cognition*, 19:587–600, 2001.

[168] Lauria, S., Bugmann, G., Kyriacou, T., and Klein, E. Mobile robot programming using natural language. *Robotics and Autonomous Systems*, 38(3–4):171–181, 2002.

[169] Lee, J., Chai, J., Reitsma, P.S.A., Hodgins, J.K., and Pollard, N.S. Interactive control of avatars animated with human motion data. In *Proc. 29th Annual Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH '02)*, pages 491–500, San Antonio, Texas, 2002.

[170] Lee, S-L., Kiesler, S., Lau, I.Y-M., and Chiu, C-Y. Human mental models of humanoid robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain, April 2005.

[171] Lewis, J. Automated lip-sync: Background and techniques. *Journal of Visualization and Computer Animation*, 2, 1991.

[172] Likhachev, M. and Arkin, R.C. Robotic comfort zones. In *Proceedings of SPIE: Sensor Fusion and Decentralized Control in Robotic Systems III Conference*, volume 4196, pages 27–41, 2000.

[173] Lim, H., Ishii, A., and Takanishi, A. Basic emotional walking using a biped humanoid robot. In *Proc. IEEE International Conference on Systems, Man and Cybernetics*, page 383, Tokyo, Japan, October 1999.

[174] Lin, L.-J. Self-improving reactive agents based on reinforcement learning, planning, and teaching. *Machine Learning*, 8:293–321, 1992.

[175] Louwerse, M.M. and Bangerter, A. Focusing attention with deictic gestures and linguistic expressions. In *Proc. XXVII Annual Conference of the Cognitive Science Society (CogSci 2005)*, Stresa, Italy, July 21–23 2005.

[176] Lungarella, M., Metta, G., Pfeifer, R., and Sandini, G. Developmental robotics: A survey. *Connection Science*, 15(4):151–190, December 2003. Special Issue on Epigenetic Robotics: Modelling Cognitive Development in Robotic Systems.

[177] Machotka, P. and Spiegel, J. *The Articulate Body*. Irvington, 1982.

[178] Mackworth, A.K. Consistency in networks of relations. *Artificial Intelligence*, 8(1):99–118, 1977.

[179] Maclin, R. and Shavlik, J.W. Creating advice-taking reinforcement learners. *Machine Learning*, 22(1–3):251–281, 1996.

[180] Madsen, R. *Animated Film: Concepts, Methods, Uses*. Interland, New York, 1969.

[181] Maes, P. Artificial life meets entertainment: Lifelike autonomous agents. *Communications of the ACM*, 38(11):108–114, November 1995.

[182] Maes, P., Darrell, T., Blumberg, B., and Pentland, A. The ALIVE system: Full-body interaction with autonomous agents. In *Proc. Computer Animation '95 Conference*, Geneva, Switzerland, April 1995.

[183] Magerko, B., Laird, J.E., Assanie, M., Kerfoot, A., and Stokes, D. AI characters and directors for interactive computer games. In *Proceedings of the 16th Conference on Innovative Applications of Artificial Intelligence (IAAI-04)*, 2004.

[184] Malcolm, D.G., Roseboom, J.H., Clark, C.E., and Fazar, W. Application of a technique for research and development program evaluation. *Operations Research*, 7:646–669, 1959.

[185] Malik, J. and Binford, T.O. Reasoning in time and space. In *Proc. International Joint Conference on Artificial Intelligence*, pages 343–345, Karlsruhe, Federal Republic of Germany, 1983.

[186] Marslen-Wilson, W., Levy, E., and Tyler, L.K. Producing interpretable discourse: The establishment and maintenance of reference. In Jarvella, R.J. and Klein, W., editors, *Speech, Place and Action: Studies in Deixis and Related Topics*. Wiley, 1982.

[187] Martinovsky, B. and Traum, D. The error is the clue: Breakdown in human-machine interaction. In *Proceedings of Error Handling in Spoken Dialogue Systems (EHSD-03)*, 2003.

[188] Matarić, M., Williamson, M., Demiris, J., and Mohan, A. Behavior-based primitives for articulated control. In *Proceedings of the Fifth International Conference of the Society for Adaptive Behavior (SAB '98)*, pages 165–170, 1998.

[189] Mateas, M. and Stern, A. A behavior language for story-based believable agents. *IEEE Intelligent Systems*, 17(4):39–47, July/August 2002.

[190] Mazor, E., Dayan, J., Averbuch, A., and Bar-Shalom, Y. Interacting multiple model methods in target tracking: A survey. *IEEE Transactions on Aerospace and Electronic Systems*, 34(1):103–123, 1998.

[191] McCarthy, J. and Hayes, P. Some philosophical problems from the standpoint of artificial intelligence. In Meltzer, B. and Michie, D., editors, *Machine Intelligence 4*, pages 463–502. Edinburgh University Press, Edinburgh, Scotland, 1969.

[192] McCrae, R.R. and Costa, P.T. Toward a new generation of personality theories: Theoretical contexts for the five-factor model. In Wiggins, J.S., editor, *Five-Factor Model of Personality*, pages 51–87. Guilford, New York, 1996.

[193] McNeill, D. *Hand and Mind: What Gestures Reveal about Thought*. University of Chicago Press, Chicago, IL, 1992.

[194] McNeill, D. and Levy, E. Conceptual representations in language activity and gesture. In Jarvella, R.J. and Klein, W., editors, *Speech, Place and Action: Studies in Deixis and Related Topics*. Wiley, 1982.

[195] Meiri, I. Combining qualitative and quantitative constraints in temporal reasoning. *Artificial Intelligence*, 87:343–385, 1996.

[196] Meltzoff, A. and Moore, M.K. Explaining facial imitation: A theoretical model. *Early Development and Parenting*, 6:179–192, 1997.

[197] Michelsen, A., Andersen, B.B., Storm, J., Kirchner, W.H., and Lindauer, M. How honeybees perceive communication dances, studied by means of a mechanical model. *Behavioral Ecology and Sociobiology*, 30(3–4):143–150, April 1992.

[198] Milota, A.D. and Blattner, M.M. Multimodal interfaces with voice and gesture input. In *Proc. International Conference on Systems, Man and Cybernetics*, pages 2760–2765, Vancouver, Canada, October 1995. IEEE.

[199] Miyauchi, D., Sakurai, A., Nakamura, A., and Kuno, Y. Active eye contact for human-robot communication. In *CHI '04: Extended Abstracts of the 2004 Conference on Human Factors in Computing Systems*, pages 1099–1102. ACM Press, April 24–29 2004.

[200] Mizoguchi, H., Sato, T., Takagi, K., Nakao, M., and Hatamura, Y. Realization of expressive mobile robot. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 1, pages 581–586, Albuquerque, New Mexico, 20–25 April 1997.

[201] Montanari, U. Networks of constraints: Fundamental properties and applications to picture processing. *Information Sciences*, 7, 1974.

[202] Moore, C. and Dunham, P.J., editors. *Joint Attention: Its Origins and Role in Development*. Lawrence Erlbaum Associates, 1995.

[203] Moore, D., Essa, I., and Hayes, M. Exploiting human actions and object context for recognition tasks. In *Proc. International Conference on Computer Vision*, Corfu, Greece, 1999.

[204] Moshkina, L. and Arkin, R.C. Human perspective on affective robotic behavior: A longitudinal study. In *Proc. International Conference on Intelligent Robots and Systems*, 2005.

[205] Mouri, T., Kawasaki, H., and Umebayashi, K. Developments of new anthropomorphic robot hand and its master slave system. In *Proc. International Conference on Intelligent Robots and Systems*, pages 3225–3230, Edmonton, Alberta, Canada, 2–6 August 2005.

[206] Murphy, R.R. Human-robot interaction in rescue robotics. *IEEE Transactions on Systems, Man and Cybernetics—Part C*, 34(2):138–153, May 2004.

[207] Murphy, R.R. and Rogers, E. Final report for DARPA/NSF study on human-robot interaction. California Polytechnic State University, San Luis Obispo, CA, September 2001.

[208] Nadel, B.A. Constraint satisfaction algorithms. *Computational Intelligence*, 5:188–224, 1989.

[209] Nagai, Y. Learning to comprehend deictic gestures in robots and human infants. In *Proc. 14th IEEE International Workshop on Robot and Human Interactive Communication (RO-MAN'05)*, pages 217–222, Nashville, TN, August 2005.

[210] Nakauchi, Y. and Simmons, R. A social robot that stands in line. In *Proc. International Conference on Intelligent Robots and Systems*, volume 1, pages 357–364, October 2000.

[211] Nass, C. and Lee, K.M. Does computer-generated speech manifest personality? Experimental test of recognition, similarity-attraction, and consistence-attraction. *Journal of Experimental Psychology, Applied*, 7(3):171–181, 2001.

[212] Neff, M. and Fiume, E. Aesthetic edits for character animation. In Breen, D. and Lin, M., editors, *Eurographics/SIGGRAPH Symposium on Computer Animation*, pages 239–244, 2003.

[213] Nehaniv, C.L. and Dautenhahn, K. Of hummingbirds and helicopters: An algebraic framework for interdisciplinary studies of imitation and its applications. In Demiris, J. and Birk, A., editors, *Learning Robots: An Interdisciplinary Approach*. World Scientific Press, 1999.

[214] Nicolescu, M.N. and Matarić, M. Learning and interacting in human-robot domains. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 31(5):419–430, 2001. Special Issue on Socially Intelligent Agents — The Human In The Loop.

[215] Nicolescu, M.N. and Matarić, M. A hierarchical architecture for behavior-based robots. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-02)*, pages 227–233, Bologna, Italy, 2002.

[216] Nicolescu, M.N. and Matarić, M. Learning movement sequences from demonstration. In *Proceedings of the International Conference on Development and Learning (ICDL-2002)*, 2002.

[217] Nofi, A.A. Defining and measuring shared situational awareness. Technical Report CRM D0002895.A1, Center for Naval Analyses, Alexandria, Virginia, November 2000.

[218] Norman, D. Emotion & design: Attractive things work better. *Interactions*, 9(4):36–42, July 2002.

[219] Nott, H.M.R. Social behavior of the dog. In Thorn, C., editor, *The Waltham Book of Dog and Cat Behavior*, pages 97–114. Pergamon Press, Oxford, 1992.

[220] Noyes, J.M. Expectations and their forgotten role in HCI. In Ghaoui, C., editor, *Encyclopaedia of Human Computer Interaction*. Idea Group Inc., Hershey, Pennsylvania, 2006.

[221] O'Regan, J.K. and Noë, A. A sensorimotor account of vision and visual consciousness. *Behavioral and Brain Sciences*, 24(5):939–1011, 2001.

[222] Orford, J. The rules of interpersonal complementarity: Does hostility beget hostility and dominance, submission? *Psychological Review*, 93:365–377, 1986.

[223] Oviatt, S. Ten myths of multimodal interaction. *Communications of the ACM*, 42(11):74–81, 1999.

[224] Oviatt, S., DeAngeli, A., and Kuhn, K. Integration and synchronization of input modes during multimodal human-computer interaction. In *Proceedings of the 1997 Conference on Human Factors in Computing Systems (CHI'97)*, pages 415–422, Atlanta, Georgia, April 1997.

[225] Pacchierotti, E., Christensen, H.I., and Jensfelt, P. Human-robot embodied interaction in hallway settings: A pilot user study. In *Proceedings of the 2005 IEEE International Workshop on Robots and Human Interactive Communication*, pages 164–171, 2005.

[226] Parke, F.I. Computer generated animation of faces. Master's thesis, University of Utah, Salt Lake City, Utah, June 1972.

[227] Parke, F.I. and Waters, K. *Computer Facial Animation*. A.K. Peters, Wellesley, Massachusetts, 1996.

[228] Paton, J.J., Belova, M.A., Morrison, S.E., and Salzman, C.D. The primate amygdala represents the positive and negative value of visual stimuli during learning. *Nature*, 439:865–870, February 2006.

[229] Pavlovic, V.I., Sharma, R., and Huang, T.S. Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):677–695, July 1997.

[230] Perkins, S. and Hayes, G. Robot shaping — principles, methods and architectures. In *Proceedings of the Workshop on Learning in Robots and Animals, AISB'96*, University of Sussex, UK, April 1–2 1996.

[231] Perlin, K. and Goldberg, A. Improv: A system for scripting interactive actors in virtual worlds. *Computer Graphics*, pages 205–218, 1996.

[232] Perzanowski, D., Schultz, A.C., Adams, W., Marsh, E., and Bugajska, M. Building a multimodal human-robot interface. *IEEE Intelligent Systems*, 16(1):16–21, 2001.

[233] Peters, R.A.II, Campbell, C.C., Bluethmann, W.J., and Huber, E. Robonaut task learning through teleoperation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2806–2811, Taipei, Taiwan, September 2003.

[234] Peters, R.A.II, Hambuchen, K.E., Kawamura, K., and Wilkes, D.M. The sensory ego-sphere as a short-term memory for humanoids. In *Proc. IEEE-RAS/RSJ Int'l Conf. on Humanoid Robots (Humanoids '01)*, pages 451–459, Tokyo, Japan, 2001.

[235] Peters, R.A.II, Wilkes, D.M., Gaines, D.M., and Kawamura, K. A software agent based control system for human-robot interaction. In *Proceedings of the Second International Symposium on Humanoid Robots*, Tokyo, Japan, October 8–9 1999.

[236] Pfeiffer, T. and Latoschik, M.E. Resolving object references in multimodal dialogues for immersive virtual environments. In *Proc. IEEE Virtual Reality Conference (VR'04)*, Chicago, IL, March 27–31 2004.

[237] Pinhanez, C.S. and Bobick, A.F. PNF calculus: A method for the representation and fast recognition of temporal structure. Technical Report 389, MIT Media Laboratory Perceptual Computing Section, Cambridge, Massachusetts, September 1996.

[238] Pinhanez, C.S. and Bobick, A.F. Human action detection using PNF propagation of temporal constraints. In *Proc. Computer Vision and Pattern Recognition*, pages 898–904, Santa Barbara, California, June 1998.

[239] Pollack, M.E., Engberg, S., Matthews, J.T., Thrun, S., Brown, L., Colbry, D., Orosz, C., Peintner, B., Ramakrishnan, S., Dunbar-Jacob, J., McCarthy, C., Montemerlo, M., Pineau, J., and Roy, N. Pearl: A mobile robotic assistant for the elderly. *AAAI Workshop on Automation as Eldercare*, August 2002.

[240] Powers, A., Kramer, A., Lim, S., Kuo, J., Lee, S-L., and Kiesler, S. Eliciting information from people with a gendered humanoid robot. In *Proceedings of the Fourteenth IEEE Workshop on Robot and Human Interactive Communication (Ro-Man'05)*, Nashville, Tennessee, August 2005.

[241] Premack, D. and Woodruff, G. Does the chimpanzee have a theory of mind? *Behavioral and Brain Sciences*, 1(4):515–526, 1978.

[242] Prinz, W. NESSIE: An awareness environment for cooperative settings. In *Proceedings of the 6th European Conference on Computer Supported Cooperative Work*, Copenhagen, Denmark, 1999.

[243] Ragni, M. and Scivos, A. Dependency calculus: Reasoning in a general point relation algebra. In Furbach, U., editor, *KI 2005: Advances in Artificial Intelligence, Proceedings of the 28th Annual German Conference on AI*, volume 3698, pages 49–63. Springer, Berlin, 2005.

[244] Ramesh, A. A metric for the evaluation of imitation. In *Proceedings of the 19th National Conference on Artificial Intelligence (AAAI 2004)*, pages 999–1000, 2004.

[245] Reeves, B. and Nass, C. *The Media Equation: How People Treat Computers, Television, and New Media Like Real People and Places.* Cambridge University Press, Cambridge, England, 1996.

[246] Renz, J. A spatial odyssey of the interval algebra: 1. Directed intervals. In *Proc. International Joint Conference on Artificial Intelligence*, pages 51–56, 2001.

[247] Rickel, J. and Johnson, W.L. Task-oriented collaboration with embodied agents in virtual worlds. In Cassell, J., Sullivan, J., Prevost, S., and Churchill, E., editors, *Embodied Conversational Agents.* MIT Press, Cambridge, Massachusetts, 2000.

[248] Rizzolatti, G., Gadiga, L., Gallese, V., and Fogassi, L. Premotor cortex and the recognition of motor actions. *Cognitive Brain Research*, 3:131–141, 1996.

[249] Robins, B., Dautenhahn, K., te Boekhorst, R., and Billard, A. Effects of repeated exposure to a humanoid robot on children with autism. In *Proc. Universal Access and Assistive Technology (CWUAAT)*, Cambridge, UK, March 2004. Springer-Verlag.

[250] Rose, C., Cohen, M., and Bodenheimer, B. Verbs and adverbs: Multidimensional motion interpolation. *IEEE Computer Graphics and Applications*, 18(5):32–40, September 1998.

[251] Roy, D., Hsiao, K.-Y., and Mavridis, N. Mental imagery for a conversational robot. *IEEE Transactions on Systems, Man and Cybernetics—Part B*, 34:1374–1383, June 2004.

[252] Sacerdoti, E.D. *A Structure for Plans and Behavior.* Elsevier, New York, 1977.

[253] Sacks, H., Schegloff, A., and Jefferson, G. A simplest systematics for the organization of turn-taking in conversation. *Language*, 50:696–735, 1974.

[254] Saksida, L.M., Raymond, S.M., and Touretzky, D.S. Shaping robot behavior using principles from instrumental conditioning. *Robotics and Autonomous Systems*, 22:231–249, 1997.

[255] Salvador, S. and Chan, P. FastDTW: Toward accurate dynamic time warping in linear time and space. In *KDD Workshop on Mining Temporal and Sequential Data*, pages 70–80, 2004.

[256] Sanders, M.S. and McCormick, E.J. *Human Factors in Engineering and Design.* McGraw-Hill, New York, 7 edition, 1993.

[257] Saunders, J., Nehaniv, C.L., and Dautenhahn, K. An examination of the static to dynamic imitation spectrum. In *Proceedings of the 3rd International Symposium on Animals and Artifacts at AISB 2005*. The Society for the Study of Artificial Intelligence and the Simulation of Behaviour (SSAISB), April 2005.

[258] Sawada, T., Takagi, T., Hoshino, Y., and Fujita, M. Learning behavior selection through interaction based on emotionally grounded symbol concept. In *Proc. IEEE-RAS/RSJ Int'l Conf. on Humanoid Robots (Humanoids '04)*, 2004.

[259] Scaife, M. and Bruner, J.S. The capacity for joint visual attention in the infant. *Nature*, 253:265–266, 1975.

[260] Scassellati, B. Theory of mind for a humanoid robot. In *First IEEE/RSJ International Conference on Humanoid Robotics*, September 2000.

[261] Schaal, S. Is imitation learning the route to humanoid robots? *Trends In Cognitive Sciences*, 3(6):233–242, 1999.

[262] Schaal, S., Ijspeert, A., and Billard, A. Computational approaches to motor learning by imitation. *Philosophical Transaction of the Royal Society of London: Series B, Biological Sciences*, 358(1431):537–547, 2003.

[263] Schaal, S., Vijayakumar, S., D'Souza, A., Ijspeert, A., and Nakanishi, J. Real-time statistical learning for robotics and human augmentation. In *Proceedings of the Tenth International Symposium on Robotics Research (ISRR 2001)*, pages 117–124, 2001.

[264] Scheeff, M., Pinto, J., Rahardja, K., Snibbe, S., and Tow, R. Experiences with Sparky, a social robot. In *Proceedings of the Workshop on Interactive Robot Entertainment*, 2000.

[265] Scholtz, J. Human-robot interactions: Creating synergistic cyber forces. In Schultz, A.C and Parker, L.E., editors, *Multi-Robot Systems: From Swarms to Intelligent Automata (Proceedings from the 2002 NRL Workshop on Multi-Robot Systems)*. Kluwer Academic Publishers, March 2002.

[266] Searle, J. Minds, brains and programs. *Behavioral and Brain Sciences*, 3:417–458, 1980.

[267] Shoham, Y. Temporal logics in AI: Semantical and ontological considerations. *Artificial Intelligence*, 33(1):89–104, 1987.

[268] Simmons, R., Bruce, A., Goldberg, D., Goode, A., Montemerlo, M., Roy, N., Sellner, B., Urmson, C., Schultz, A.C., Adams, W., Bugajska, M., MacMahon, M., Mink, J., Perzanowski, D., Rosenthal, S., Thomas, S., Horswill, I., Zubek, R., Kortenkamp, D., Wolfe, B., Milam, T., and Maxwell, B. GRACE and GEORGE: Autonomous robots for the AAAI robot challenge. In *Workshop Paper for AAAI 2003*, 2003.

[269] Simpson, B.S. Canine communication. *Veterinary Clinics of North America: Small Animal Practice*, 27(3):445–464, 1997.

[270] Singh, S.P. Transfer of learning across sequential tasks. *Machine Learning*, 8:323–339, 1992.

[271] Skinner, B.F. *Science and Human Behavior*. Macmillan, New York, 1953.

[272] Slater, M., Howell, J., Steed, A., Pertaub, D.-P., and Gaurau, M. Acting in virtual reality. In *Proceedings of the Third International Conference on Collaborative Virtual Environments (CVE 2000)*, pages 103–110, San Francisco, California, September 10–12 2000.

[273] Smith, C. Behavior adaptation for a socially interactive robot. Master's thesis, KTH Royal Institute of Technology, Stockholm, Sweden, 2005.

[274] Snibbe, S., Scheeff, M., and Rahardja, K. A layered architecture for lifelike robotic motion. In *Proceedings of the 9th International Conference on Advanced Robotics*, Tokyo, Japan, 1999.

[275] Spence, K.W. Experimental studies of learning and higher mental processes in infra-human primates. *Psychological Bulletin*, 34:806–850, 1937.

[276] Stiehl, W.D. Sensitive skins and somatic processing for affective and sociable robots based upon a somatic alphabet approach. Master's thesis, MIT Media Laboratory, Cambridge, Massachusetts, 2005.

[277] Strassman, S. *Desktop Theater: Automatic Generation of Expresssive Animation*. PhD thesis, MIT Media Laboratory, Cambridge, Massachusetts, June 1991.

[278] Strobel, M., Illmann, J., Kluge, B., and Marrone, F. Using spatial context knowledge in gesture recognition for commanding a domestic service robot. In *Proc. 11th IEEE Workshop on Robot and Human Interactive Communication (RO-MAN'02)*, pages 468–473, Berlin, Germany, September 25–27 2002.

[279] Taddeo, M. and Floridi, L. Solving the symbol grounding problem: A critical review of fifteen years of research. *Journal of Experimental and Theoretical Artificial Intelligence*, 17(4):419–445, 2005.

[280] Taranovsky, D. Guided control of intelligent virtual puppets. Master's thesis, University of Toronto, Toronto, Canada, 2001.

[281] te Boekhorst, R., Walters, M., Koay, K.L., Dautenhahn, K., and Nehaniv, C. A study of a single robot interacting with groups of children in a rotation game scenario. In *Proc. 6th IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA 2005)*, Espoo, Finland, June 2005.

[282] Thorisson, K.R. *Communicative Humanoids: A Computational Model of Psychosocial Dialogue Skills*. PhD thesis, MIT Media Laboratory, Cambridge, Massachusetts, 1996.

[283] Thorpe, W.H. *Learning and Instinct in Animals*. Methuen, London, United Kingdom, 1956.

[284] Thrun, S., Bennewitz, M., Burgard, W., Cremers, A., Dellaert, F., Fox, D., Haehnel, D., Rosenberg, C., Roy, N., Schulte, J., and Schulz, D. MINERVA: A second generation mobile tour guide robot. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 1999.

[285] Tomasello, M. Cultural transmission in the tool use and communicatory signalling of chimpanzees? In Parker, S.T. and Gibson, K.R., editors, *Comparative Developmental Psychology of Language and Intelligence in Primates*, pages 274–311. Cambridge University Press, 1990.

[286] Tomlinson, W.M. Using human acting skill to measure empathic value in heterogeneous characters. In *Third International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-04), Workshop on Empathic Agents*, New York, New York, 2004.

[287] Tu, X. and Terzopoulos, D. Artificial fishes: Physics, locomotion, perception, behavior. In *Proc. 21st Annual Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH '94)*, pages 43–50, Orlando, Florida, 1994.

[288] van Beek, P. Approximation algorithms for temporal reasoning. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence (IJCAI-89)*, pages 1291–1296, Detroit, Michigan, 1989.

[289] van Breemen, A.J.N. Bringing robots to life: Applying principles of animation to robots. In *Proceedings of the 2004 Conference on Human Factors in Computing Systems (CHI'04)*, Vienna, Austria, April 24–29 2004.

[290] Vertegaal, R., Velichkovsky, B., and van der Veer, G. Catching the eye: Management of joint attention in cooperative work. *SIGCHI Bulletin*, 29(4), 1997.

[291] Vilain, M. and Kautz, H. Constraint propagation algorithms for temporal reasoning. In *Proceedings of the 4th National Conference on Artificial Intelligence (AAAI 1986)*, pages 297–326, Philadelphia, Pennsylvania, 1986.

[292] Vilain, M., Kautz, H., and van Beek, P. Constraint propagation algorithms for temporal reasoning: A revised report. In Weld, D.S. and de Kleer, J., editors, *Readings in Qualitative Reasoning about Physical Systems*, pages 373–381. Morgan Kaufmann, San Mateo, California, 1990.

[293] Viola, P. and Jones, M. Rapid object detection using a boosted cascade of simple features. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 511–518, Kauai, HI, December 2001.

[294] Voyles, R.M. and Khosla, P.K. Tactile gestures for human/robot interaction. In *Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems (IROS '95)*, volume 3, pages 7–13, Pittsburgh, Pennsylvania, 1995.

[295] Wada, K., Shibata, T., Saito, T., and Tanie, K. Psychological and social effects of robot assisted activity to elderly people who stay at a health service facility for the aged. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3996–4001, Taipei, Taiwan, September 2003. IEEE.

[296] Walters, M.L., Dautenhahn, K., Koay, K.L., Kaouri, C., te Boekhorst, R., Nehaniv, C.L., Werry, I., and Lee, D. Close encounters: Spatial distances between people and a robot of mechanistic appearance. In *Proceedings of the 5th IEEE-RAS International Conference on Humanoid Robots (Humanoids '05)*, pages 450–455, Tsukuba, Japan, 5–7 December 2005.

[297] Walters, M.L., Dautenhahn, K., te Boekhorst, R., Koay, K.L., Kaouri, C., Woods, S., Nehaniv, C.L., Lee, D., and Werry, I. The influence of subjects' personality traits on personal spatial zones in a human-robot interaction experiment. In *Proceedings of IEEE Ro-Man*

2005, 14th Annual Workshop on Robot and Human Interactive Communication, pages 347–352, Nashville, Tennessee, 13–15 August 2005. IEEE Press.

[298] Watson, D., Clark, L.A., and Tellegen, A. Development and validation of brief measures of positive and negative affect: The PANAS scales. *Journal of Personality and & Social Psychology*, 56(6):1063–1070, 1998.

[299] Weitz, S. *Nonverbal Communication: Readings With Commentary*. Oxford University Press, 1974.

[300] Werger, B. Profile of a winner: Brandeis University and Ullanta Performance Robotics "Robotic Love Triangle". *AI Magazine*, 19(3):35–38, 1998.

[301] Weyhrauch, P. *Guiding Interactive Drama*. PhD thesis, Department of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, 1997.

[302] Whiten, A. The dissection of socially mediated learning. (forthcoming; pre-publication manuscript kindly provided by Andrew Whiten), 2002.

[303] Whiten, A. and Ham, R. On the nature and evolution of imitation in the animal kingdom: Reappraisal of a century of research. In Slater, P.J.B., Rosenblatt, S., Beer, C., and Milinsky, M., editors, *Advances In The Study Of Behavior*, pages 239–283. Academic Press, New York, 1992.

[304] Wood, D. Social interaction as tutoring. In Bornstein, M.H. and Bruner, J.S., editors, *Interaction in Human Development*, pages 59–80. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1989.

[305] Wood, D., Bruner, J.S., and Ross, G. The role of tutoring in problem-solving. *Journal of Child Psychology and Psychiatry*, 17:89–100, 1976.

[306] Wood, G. *Edison's Eve: A Magical History of the Quest for Mechanical Life*. Alfred A. Knopf, 2002.

[307] Yamasaki, N. and Anzai, Y. Active interface for human-robot interaction. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3103–3109, 1996.

[308] Yan, C., Peng, W., Lee, K.M., and Jin, S. Can robots have personality? An empirical study of personality manifestation, social responses, and social presence in human-robot interaction. In *54th Annual Conference of the International Communication Association*, 2004.

[309] Yanco, H.A. and Drury, J.L. "Where am I?" Acquiring situation awareness using a remote robot platform. In *Proceedings of the IEEE Conference on Systems, Man and Cybernetics*, The Hague, Netherlands, October 2004.

[310] Yang, J. and Waibel, A. A real-time face tracker. In *Proc. 3rd Workshop on Applications of Computer Vision*, pages 142–147, Sarasota, Florida, USA, 1996. IEEE.

[311] Yuille, A.L., Cohen, D.S., and Hallinan, P.W. Feature extraction from faces using deformable templates. In *Proc. Computer Vision and Pattern Recognition*, San Diego, CA, June 1989.

[312] Zecca, M., Roccella, S., Carrozza, M.C., Cappiello, G., Cabibihan, J.-J., Dario, P., Takanobu, H., Matsumoto, M., Miwa, H., Itoh, K., and Takanishi, A. On the development of the emotion expression humanoid robot WE-4RII with RCH-1. In *Proc. IEEE-RAS/RSJ Int'l Conf. on Humanoid Robots (Humanoids '04)*, Los Angeles, California, November 2004.

[313] Zelek, J.S. Human-robot interaction with minimal spanning natural language template for autonomous and tele-operated control. In *Proceedings of the 1997 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'97)*, volume 1, pages 299–305, Grenoble, France, 7–11 September 1997.

[314] Zentall, T.R. Imitation in animals: Evidence, function and mechanisms. *Cybernetics and Systems*, 32(1–2):53–96, 2001.

[315] Ziemke, T. Rethinking grounding. In Riegler, A., Peschl, M., and von Stein, A., editors, *Understanding Representation in the Cognitive Sciences*, pages 177–190. Plenum Press, New York, 1999.

[316] Zue, V., Seneff, S., Glass, J.R., Polifroni, J., Pao, C., Hazen, T.J., and Hetherington, L. JUPITER: A telephone-based conversational interface for weather information. *IEEE Transactions on Speech and Audio Processing*, 8(1):85–96, 2000.