THE UNIVERSITY OF QUEENSLAND

AUSTRALIA

Evolving spatial and temporal lexicons across different cognitive architectures

Scott Heath

B.E. (Hons.), B. Inf. Tech.

A thesis submitted for the degree of Doctor of Philosophy at

The University of Queensland in 2015

School of Information Technology and Electrical Engineering

Abstract

Communication between mobile robots requires a transfer of symbols, where each symbol signifies a meaning. However, in typical applications, meaning has been ascribed to the symbols by the engineers that have programmed the robots. This thesis explores an alternative: the use of algorithms and representations that allow mobile robots to evolve a shared set of symbols where the meanings of the symbols are derived from the robots' sensors and cognition.

Mobile robots have two important properties that affect the learning of symbols, i) that they are capable of locomotion through space over time; and ii) that they come in many different configurations with different architectures. Previous work has demonstrated that mobile robots can learn shared lexicons to describe space through perceptual referents and referents grounded in cognitive maps. However, open questions remain as to how mobile robots can learn to communicate using temporal terms, and how learning lexicons is affected by different cognitive architectures.

The major research question addressed in this thesis is *how can mobile robots develop spatial and temporal lexicons across different cognitive architectures?* Three facets of language learning are considered particularly important for robots with different cognitive architectures: i) the ability to ground terms in cognition; ii) the ability to ground identical terms in different sensors and cognition for each robot; and iii) the ability to handle referential uncertainty - the difficulty of linking words to meanings within ambiguous contexts. Pairs of mobile robots are used to develop lexicons for spatial and temporal terms and to study each of these abilities. The terms developed by the robots are tested by organizing spatial and temporal tasks and extended to additional terms through grounding transfer.

In this thesis, language learning is studied within a framework defined by Peirce's semiotic triangle and building on previous Lingodroid studies. Conversations between robots are used to socially ground symbols within the robots' spatial and temporal cognition. Distributed lexicon tables are used to store links between words and meanings. As the lexicons evolve the words are analyzed for immediate usability, and the final lexicons are analyzed for coherence.

Four studies to analyze different aspects of lexicon learning were completed. Study I addressed the aims of learning duration terms using mobile robots and using grounded spatial and temporal language together to perform joint tasks. Identical mobile robots were used to ground terms for time in durations using clocks (time since the last meeting). The robots were able to develop coherent lexicons, and successfully organize future meetings using learned terms.

Study II addressed the aim of learning event-based temporal terms using mobile robots. Identical mobile robots were used to ground terms for time in sunlight levels (time of day). The robots required the ability to ground terms in features formed from a brightness level and its derivative. Again the robots were

able to develop coherent lexicons and organize meetings, handling changing daylight cycles throughout a year.

Study III addressed the aim of learning spatial terms across different cognitive architectures. Robots with different sensors and spatial cognition were used to ground spatial terms within their different spatial representations. These spatial terms could then be used to bootstrap terms for distances and directions, unifying the two robots' different spatial terms into identically represented higher-level terms. The robots were able to develop coherent lexicons for distances and directions. This suggests that the underlying spatial terms – grounded in different spatial sensors and cognition – were also coherent.

Study IV addressed the aim of resolving uncertainty using cross-situational learning. The same pair of robots within a simulator were used to ground terms for space and time but in uncertain conditions where the feature of interest was not communicated *a priori*. The robots in this study used information metrics with cross-situational learning to decide when to link a word and meaning. Cross-situational learning was compared to the lexicon learning from the previous studies on learning time and usability. Results showed that the robots were capable of learning coherent lexicons despite the uncertainty, although with an increase in learning time and a decrease in immediate usability.

From the four completed studies, three major conclusions have been drawn. Firstly, the coherence of the lexicons in each study demonstrate that it is possible to i) learn terms for durations grounded in clock time; ii) learn terms for times of day grounded in sunlight levels; iii) ground distances and directions in different underlying spatial representations; and iv) ground spatial and temporal lexicons across different cognitive architectures and achieve communicative success.

The second major conclusion is the set of changes to the Lingodroids framework that are required for handling different learning tasks. For a system such as Lingodroids, the ability to generalize the core of the framework over multiple scenarios is one of the most important characteristics. This thesis demonstrates that the same distributed lexicon tables, conversations, categorization and generalization can be used across all study conditions. However, certain aspects of the Lingodroids framework do not generalize, and these aspects represent observations about the key differences between each of the learning conditions. The required changes include referents for new representations of time and space, cross-situational learning, and temporal cognition.

The third major conclusion is an expansion of the nature of semiotics. Traditionally a symbol was linked to a referent in the environment through a private representation. However, for robots with different cognitive architectures, a shared symbol may be linked to multiple different private representations, and to multiple different sensors before linking back to a referent in the environment.

Declaration by author

This thesis is composed of my original work, and contains no material previously published or written by another person except where due reference has been made in the text. I have clearly stated the contribution by others to jointly-authored works that I have included in my thesis.

I have clearly stated the contribution of others to my thesis as a whole, including statistical assistance, survey design, data analysis, significant technical procedures, professional editorial advice, and any other original research work used or reported in my thesis. The content of my thesis is the result of work I have carried out since the commencement of my research higher degree candidature and does not include a substantial part of work that has been submitted to qualify for the award of any other degree or diploma in any university or other tertiary institution. I have clearly stated which parts of my thesis, if any, have been submitted to qualify for another award.

I acknowledge that an electronic copy of my thesis must be lodged with the University Library and, subject to the policy and procedures of The University of Queensland, the thesis be made available for research and study in accordance with the Copyright Act 1968 unless a period of embargo has been approved by the Dean of the Graduate School.

I acknowledge that copyright of all material contained in my thesis resides with the copyright holder(s) of that material. Where appropriate I have obtained copyright permission from the copyright holder to reproduce material in this thesis.

Publications during candidature

1. Heath, S., Ball, D. and Wiles, J. (In Press, submitted 2015). Lingodroids: Cross-situational learning for episodic elements. In Cangelosi, A. (Ed.). *IEEE Transactions on Autonomous Mental Development*.

2. Gibson, T. T. (A.), Heath, S., Quinn R. P., Lee, A. H., Arnold, J. T., Sonti, T. S., Whalley, A., Shannon, G. P., Song, B. T., Henderson, J. A. and Wiles, J. (2014). Event-based visual data sets and prediction tasks for spiking networks. In Wermter, S., Weber, C., Duch, W., Honkela, T., Koprinkova-Hristova, P., Magg, S., Palm, G., Villa, A.E.P. (Eds.). *Proceedings of 2014 International Conference on Artificial Neural Networks*, Hamburg, Germany.

3. Heath, S., Ball, D., Schulz, R. and Wiles, J. (2013). Communication between Lingodroids with different cognitive capabilities. In Parker, L. (Ed.). *Proceedings of the International Conference on Robotics and Automation*, Karlsruhe, Germany.

4. Ball, D., Heath, S., Wiles, J., Wyeth, G., Corke, P. and Milford, M. (2013). OpenRatSLAM: An open source brain-based SLAM system. In Pantofaru, C., Chitta, S., Gerkey, B., Rusu, R., Smart, W. D. and Vaughan, R. (Eds.). *Autonomous Robots*, 34(3):149-176.

5. Wiles, J., Heath., S., Ball, D., Quinn, L. and Chiba, A. (2012). Rat meets iRat. In *Proceedings of the 2012 International Conference on Development and Learning and Epigenetic Robotics*, San Diego, USA.

6. Heath, S., Schulz, R., Ball, D. and Wiles, J. (2012). Long Summer Days: Grounding words in the temporal cycles of real world events. In Harris Jr., F. C., Krichmar, J., Siegelmann, H. and Wagatsuma, H. (Eds.). *IEEE Transactions on Autonomous Mental Development*, 4(3):192-203.

7. Heath, S., Schulz, R., Ball, D. and Wiles, J. (2012). Lingodroids: Learning terms for time. In Parker, L. (Ed.). *Proceedings of the International Conference on Robotics and Automation*. Saint Paul, MN, USA.

8. Heath, S., Cummings, A., Wiles, J. and Ball, D. (2011). A rat in the browser. In Drummond, T. and Li, W. H. (Eds.). *Proceedings of the 2011 Australasian Conference on Robotics and Automation*. Melbourne, Australia.

Publications included in this thesis

Key					
D = design	C = coding	I = implementation	W = writing	A = analysis	E = editing

1. Heath, S., Ball, D. and Wiles, J. (In Press, submitted 2015). Lingodroids: Cross-situational learning for episodic elements. Submitted to Cangelosi, A. (Ed.). *IEEE Transactions on Autonomous Mental Development*. This publication is reproduced within Chapter 7.

Contributor	Statement of contribution
Scott Heath (Candidate)	D (50%), C (100%), I (100%), A (100%), W (70%), E (40%)
David Ball	D (25%), E (20%)
Janet Wiles	D (25%), W (30%), E (40%)

2. Heath, S., Ball, D., Schulz, R. and Wiles, J. (2013). Communication between Lingodroids with different cognitive capabilities. In Parker, L. (Ed.). *Proceedings of the International Conference on Robotics and Automation*, Karlsruhe, Germany. This publication is reproduced within Chapter 6.

Contributor	Statement of contribution
Scott Heath (Candidate)	D (40%), C (80%), I (100%), A(50%), W (20%), E(25%)
David Ball	D (20%), C (20%), E (25%)
Ruth Schulz	D (20%), A (50%), W (60%), E (25%)
Janet Wiles	D (20%), W (20%), E (25%)

3. Heath, S., Schulz, R., Ball, D. and Wiles, J. (2012). Long Summer Days: Grounding words in the temporal cycles of real world events. In Harris, Jr., F. C., Krichmar, J., Siegelmann, H. and Wagatsuma, H. (Eds.). *IEEE Transactions on Autonomous Mental Development*, 4(3):192-203. This publication is reproduced within Chapter 5.

Contributor	Statement of contribution
Scott Heath (Candidate)	D (33%), C (80%), I (100%), A (100%), W (20%), E (40%)
Ruth Schulz	D (33%), W (40%)
David Ball	C (20%), E (20%)
Janet Wiles	D (33%), W (40%), E (40%)

4. Heath, S., Schulz, R., Ball, D. and Wiles, J. (2012). Lingodroids: Learning terms for time. In Parker, L. (Ed.). *Proceedings of the International Conference on Robotics and Automation*. Saint Paul, MN, USA. This publication is reproduced within Chapter 4.

Contributor	Statement of contribution
Scott Heath (Candidate)	D (25%), C (80%), I (80%), A (20%), W (5%), E (25%)
Ruth Schulz	D (25%), I (20%), A (80%), W (85%), E (25%)
David Ball	D (25%), C (10%), W (5%), E (25%)
Janet Wiles	D (25%), W (5%), E (25%)

5. Ball, D., Heath, S., Wiles, J., Wyeth, G., Corke, P. and Milford, M. (2013). OpenRatSLAM: An open source brain-based SLAM system. In Pantofaru, C., Chitta, S., Gerkey, B., Rusu, R., Smart, W. D. and Vaughan, R. (Eds.). *Autonomous Robots*, 34(3):149-176. This publication is reproduced as Appendix A.

Contributor	Statement of contribution
David Ball	D (25%), C (45%), I (60%), A (50%), W (30%), E (20%)
Scott Heath (Candidate)	D (15%), C (45%), I (20%), A (20%), W (10%)
Janet Wiles	D (10%), W (10%), E (20%)
Gordon Wyeth	D (25%), W (10%), E (20%)
Peter Corke	W (10%), E (20%)
Michael Milford	D (25%), C (10%), I (20%), A(30%), W (30%), E (20%)

Contributions by others to the thesis

- The Lingodroids project provided the foundations for this thesis. The Lingodroids project was created by Ruth Schulz, Janet Wiles and Gordon Wyeth.
- The iRat platform used within this thesis was conceived by Janet Wiles and developed by David Ball.
- Janet Wiles, David Ball and Ruth Schulz all contributed to the ideas within this thesis and co-authored on the papers included in this thesis.
- Janet Wiles wrote the grants that funded this thesis and associated work.

Statement of parts of the thesis submitted to qualify for the award of another degree

None

Acknowledgements

PhDs are never just the work of an individual – there are many people who contributed to the ideas in this thesis, to my development and education as a researcher, and to my sanity. I am particularly indebted to the following people:

My supervisors, Janet Wiles and David Ball, for their enthusiasm for my research and their roles as teachers and mentors to me. I was lucky enough to have a cognitive scientist and roboticist, which meant my publications had both a long methodology and a long discussion. I appreciated Janet's enthusiasm for science, her mentoring on the "dynamics of research", her ability to turn small ideas into science worth writing about and the huge amount of time that she spent on re-structuring and rethinking studies and discussions in our publications. I appreciated David's early mentoring when I was moving from a research assistant to a PhD candidate and his continued support and efforts after switching universities.

Ruth Schulz, for helping me get started on the Lingodroids project, working with me on several papers, and for a horrendous, all-night effort for the first paper we worked on together.

The CIS group: Amy, for being forced to share a lab and a trip to the US with me; Rachael, who was forced to share part of a desk with me; and also Dan A., Marcus, and later James, Brian, Dimitri, Lydia and David T. for interesting discussion and feedback on ideas.

Contributors to publications and technology during my thesis: Angus Cummings, Michael Milford, Gordon Wyeth, Peter Corke and Sam Brian.

Contributors to proof-reading and editing of this thesis: Janet Wiles, David Ball, Alana Campbell, Lydia Byrne, and *contributors to proof-reading and editing of the publications in the thesis:* James Henderson, Dimitri Klimenko, Lydia Byrne, David Tingley and anonymous reviewers.

Thesis examiners, for their constructive feedback, and Dan Angus for agreeing to be chair of examiners.

TDLC, particularly the Chiba lab at UCSD.

Gavin Taylor and Will Maddern, for interesting discussion during our time living together. Gavin for forcing me to maintain good shopping and cooking habits, and Will for discussing ideas about symbol grounding that became influential to my thesis and publications.

Dad, Mum, Adam, Mia and Michael, for putting up with me and looking out for me financially and of course *Kerrie*, for putting up with me.

My friends for providing much needed distractions from my PhD. Particularly Michael, Adi and Rob, my tennis partners; Chris, and my section in feds Ness, Nathaniel and Anton for putting up with my often mediocre Baritone playing for many years; Dan C., Betsy and James S.

Acknowledgement of Grants

This Ph.D was funded under a University of Queensland Research Scholarship. The projects that this Ph.D and thesis contributed to were funded by an NSF Temporal Dynamics of Learning Center grant for the iRat robot platform, and Australian Research Council Discovery Projects, Special Research Centre of Thinking Systems and Centre of Excellence for the Dynamics of Language for the Lingodroids studies.

I am very grateful for these grants, which have enabled me the choice of undertaking a PhD and which have supported my work and travel throughout the PhD.

Keywords

symbol, grounding, referential, uncertainty, robot, agent, cognition, language, space, time

Australian and New Zealand Standard Research Classifications (ANZSRC)

ANZSRC code: 080101, Adaptive Agents and Intelligent Robotics, 100%

Fields of Research (FoR) Classification

FoR code: 0801, Information Sciences, 100%

Table of contents

List of figures

List of tables

List of terminology

Cognitive architecture	An agent's intelligence, formed from the agent's sensors, cognition and embodiment
Embodiment	An agent's physical body
Gmapping	A *SLAM* system based on particle filters
iRat	A rat-sized mobile robot platform developed at UQ for robot-rodent interactions. Short for intelligent rat animat technology.
IR sensors	Infra-red sensors, used for range-finding.
KDE	Kernel Density Estimator, a non-parametric learning technique, which dynamically forms distributions by convolving a kernel with instances.
Occupancy grid	A map represented as a 2D array with the value of elements indicating whether a block is free or occupied
RatSLAM	A SLAM system based on the rodent hippocampus
ROS	Robot Operating System, a software middleware for robots. ROS provides communication and abstraction for coupled software applications.
SLAM	Simultaneous Localization and Mapping
Topological map	A map represented as a graph, i.e. with nodes and edges
XSL	Cross-situational learning, learning by finding invariants from presentations of the same word with different contexts (or situations).

Chapter 1

Introduction

For mobile robots, communication with humans and with each other underlies many abilities that are expected in the future, in particular, the ability to collaborate to perform tasks. In current studies into collaboration between mobile robots, communication requires streams of symbols where programmers provide both the meanings of symbols and the knowledge that allows robots to transform symbols between different sensors and cognition (Simmons et al., 2000; Simmons et al., 2000). Two robots that perceive and process the world differently could exchange the co-ordinates of an object as binary-encoded decimal numbers (a symbol), but it is the programmers who write the code to translate from one robot's world view to decimal numbers and then to another robot's world view. Encoding these transforms *a priori* allows programmers to control exactly what mobile robots can communicate about and exactly how the symbols are processed. However, there are disadvantages: the communication implementations do not scale to large numbers of different robots (a different protocol is required for every pair of robots with different cognitive architectures) and the robots are unable to evolve meanings of symbols to track changes in embodiment, environment and peers.

An alternative to innate meanings and transforms is for robots to autonomously develop and associate symbols with their own sensors and cognition. Peirce's semiotic triangle provides a conceptual framework for agents learning links between the symbols that make up a language and their associated meanings (Ogden and Richards, 1923), a process called *symbol grounding* (Harnad, 1990). Symbol grounding enables robots to develop flexible mappings between symbols and their own sensors and cognition, and maintain and update these mappings over time to accommodate for changes in language and the environment.

Mobile robots have two important properties that must be addressed in order to enable effective symbol grounding. Firstly, these robots must have the ability to move through space over time. Space and time form the foundations of human cognition (Boroditsky, 2000; Levinson, 2003) and a similar claim has been made for mobile robots' cognition (Schulz et al., 2011a). Spatial and temporal symbols are therefore required to allow mobile robots to specify tasks that rely on locomotion.

Secondly, mobile robots are not all manufactured the same – different tasks have different hardware requirements, software requirements and operating requirements. To accommodate different tasks, mobile robots need to specialize in different areas. To communicate, these robots therefore need the ability to simultaneously ground the same symbols in their different cognitive architectures.

Previous robot language learning studies demonstrate how robots can learn lexicons for space grounded

directly in perception (Steels, 1995, 1999; Roy, 2002a), and for space and time grounded in mental maps (Jung and Zelinsky, 2000; Schulz et al., 2011a, 2011b; Spranger, 2012); however temporal naming and communication is limited in the types of times that can be learned, and aside from Jung and Zelinsky (2000), none of these studies address different cognitive architectures. Jung and Zelinsky demonstrate how robots with different sensors can learn a simple spatial language to help with a vacuuming task; however, their robots have identical cognition and their language learning framework has limitations in how their robots share attention, generalize words and use words.

Human-robot interaction studies that look at learning natural language have to deal with the different cognitive architectures of human and robot (Steels and Kaplan, 2002; Kollar et al., 2010), but these studies do not share the same perspective on language learning. Humans completely define the lexicons in these studies, which is ideal for teaching robots natural language, but loses the important dynamics of language bootstrapping: robots introducing terms into the language, generalizing terms and creating terms for their own representations. Humans cannot be analyzed in the same way that robots can – a robots' mental development and representations can be examined and visualized.

Finally, there are several studies that look only at the ambiguity between linking a word to a set of candidate meanings. These studies are divided between mathematically modeled examples that do not refer to grounded language (Smith et al., 2006; Fontanari et al., 2009) and frameworks referring to language grounded in perception (Roy, 2002b). None of these studies look at grounding in cognitive processes or cognitive differences between robots.

There is currently no comprehensive framework for language learning on mobile robots that captures all of the following criteria:

- grounding spatial and temporal terms within cognitive processes;
- grounding symbols in different underlying sensors and representations; and
- resolving the uncertainty in the links between words and meanings.

Previous studies have particular limitations in temporal language learning, grounding symbols in different cognition and dealing with the uncertainty between cognitively grounded words and meanings. In order to address these limitations, a new framework is required that is capable of:

- extending temporal language learning to new types and tasks;
- exploring language learning on robots with different spatial sensors and representations; and
- using statistical learning to resolve the uncertainty between words and meanings grounded in cognition.

To allow the framework to be analyzable, applicable to real mobile robots and capable of tracking changes in language and cognition, the following characteristics are also required:

- mobile robots bootstrapping language – as described above, human robot communication omits important dynamics;

- running on real robots, or high-fidelity simulation – real world noise provides part of the motivation for categorization and forming cognitive representations; and

- capable of running online – online operation allows robots to continuously update their lexicons to track changes in language.

The aim of this thesis is to develop and analyze a new flexible framework that meets the capabilities and characteristics described above, and test the framework on its ability to learn through each different capability. Success of the framework was measured by tests for coherence between the lexicons of different agents, learning times, coverage of features that can be named and usefulness of the lexicons in performing tasks. The analysis of the framework looked at sufficiencies and requirements to achieve the capabilities listed above and how the framework fits into existing ideas on symbol grounding and semiotics.

The developed framework was modeled on the Lingodroids framework – a framework that has already successfully demonstrated grounding of spatial terms in cognitive maps (Schulz et al., 2011a). The studies in this thesis extend lexicon learning capabilities to allow learning of terms: i) for new groundings of time, including durations (Study I) and cyclic time (Study II), ii) across different cognitive architectures (Study III), and iii) with uncertainty about links between words and meanings (Study IV).

1.1 Thesis outline

This thesis is a *thesis by publication* and includes an introduction, a literature review, an extended methodology, four self-contained publications and a general discussion. The publication chapters include the re-formatted published material with a linking prologue explaining the importance of the publication to the thesis. The thesis contains the following chapters (publications are marked with an asterisk):

Literature Review: Symbol grounding for autonomous robots and other agents (Chapter 2): This chapter reviews the symbol grounding problem and proposed solutions within both physical symbol systems and for communicating agents, starting from the thought experiments of Searle (1980) and Harnad (1990). In 2008, Steels claimed that the symbol grounding problem was solved (Steels, 2008); however, the proposed solution does not address the grounding of symbols beyond just perceptual referents. This chapter describes the types of grounding that lie outside Steels' proposed solution. Finally different frameworks for lexicon learning are evaluated on their ability to meet a criteria for learning using mobile robots.

Lingodroids 2: A new framework for autonomous lexicon learning (Chapter 3): This chapter presents the methodology that is used in all of the following studies to develop a new lexicon learning framework. Extra detail is provided above that included in the following self-contained publications. This extra detail is not required for understanding the following chapters, but it may be useful for implementing the study. This chapter covers the robot platforms, control, communication and environments; the architecture of a new version of Lingodroids developed for the robot platforms; and OpenRatSLAM – a RatSLAM implementation that was developed alongside this thesis.

Study I – Lingodroids: Learning terms for time (Chapter 4)*: Temporal cognition and communication are important to mobile robots. The studies in Chapter 4 explored learning duration concepts in a lexicon learning framework. Previous Lingodroid studies tested temporal lexicons using coherence; however, this measure does not take into account how terms are used. The studies in Chapter 4 introduced a new language game providing robots with the ability to schedule meetings using previously learned spatial and temporal terms.

Study II – Long summer days: Grounded learning of words for the uneven cycles of real world events (Chapter 5)*: Different forms of time are better suited to different tasks. The studies in Chapter 5 explored the learning of terms for times of day, terms that are grounded in the uneven cycles of sunlight. These terms are particularly suited to describing events that are best specified relative to the sun's movement during a day, instead of clock time. The ability to organize meetings was extended to use time of day terms for these studies.

Study III – Communication between Lingodroids with different cognitive architectures (Chapter 6)*: Mobile robots with different cognitive architectures need to be able to compensate for their cognitive differences to communicate successfully. The studies in Chapter 6 explored the grounding of spatial terms across robots with different spatial sensors and cognition. The studies demonstrated how spatial terms can be grounded in different spatial representations. The spatial terms are then evaluated using *grounding transfer* to develop distance and direction terms. Coherence of the transferred terms is then measured to provide an estimate of lexicon coherence.

Study IV – Lingodroids: Cross-situational learning of episodic elements (Chapter 7)*: Resolving the uncertainty between terms and their meanings is an important part of language learning. The studies in Chapter 7 introduced cross-situational learning (XSL) to the new Lingodroids framework to compensate for uncertainty between term and meaning when learning spatial and temporal terms. These studies compared learning with XSL to learning without XSL and incorporated the extensions described by all of the previous chapters, creating an integrated framework to address the problems of grounding language using mobile robots.

General discussion (Chapter 8): This chapter presents a general discussion of the contributions and implications of this thesis. The main contribution is to develop and analyze a framework for enabling grounded, autonomously-learned communication for mobile robots with different cognitive architectures. This thesis has implications for XSL, referential uncertainty, symbol grounding and semiotics.

Chapter 2

Literature Review: Symbol grounding for autonomous robots and other agents

Semiotics – the study of symbols and their associations and meanings – has become important to the fields of artificial intelligence, human-robot interactions, natural language querying and text analytics. For robots to be able to learn to communicate, they need to be able to produce and comprehend symbols that refer to aspects of the robot's self, peers and environment. This chapter provides a review of artificial symbol grounding and the processes and representations of symbol grounding that are required or typically implemented.

The ideas that underlie semiotics can be summarized using Peirce's semiotic triangle (Ogden and Richards, 1923). The semiotic triangle describes the link between symbol and meaning and was suggested by Peirce in the late 1800's with a version similar to the modern form published in 1923. The semiotic triangle and the framework it describes were based on human communication and predate the notions of *artificial intelligence* (AI) that were inspired by the invention of computers and autonomous robots in the 1940's and 50's.

The term *symbol grounding,* to describe the links between a computational symbol and its meaning, emerged from questions about the relationships between computational symbols and an agent's embodiment and environment (Searle, 1980; Harnad, 1990; Brooks, 1990; Ziemke, 2003). Early physical symbol systems used symbols with arbitrary names that were essentially more meaningful to experimenters than to the agents. Later studies concluded that in order for an agent to understand symbols, the symbols need to be linked to the environment through the agent's perception (Harnad, 1990; Barsalou, 1999; Sun, 2000); or that agents exhibit more intelligent behavior if they are implemented without symbols and instead use signals directly from their environment (Brooks, 1991).

In this background and literature review chapter, the symbol grounding problem is described and important concepts and frameworks are addressed from the perspective of application to mobile robots. Different methods for symbol grounding share common features, such as the data-structures required to link words to meanings, or the ability to categorize similar perceptible entities into equivalent groups. This chapter reviews the symbol grounding literature, with a focus on mobile robots and spatial and temporal languages. The chapter is organized into i) a comparison of communicative symbols vs. physical symbols (Section 2.1), ii) background on symbol grounding (Section 2.2), iii) background on more difficult problems

in symbol grounding including that of learning spatial and temporal terms (Section 2.3), and finally iv) a review of notable frameworks that perform symbol grounding (Section 2.4). The literature review is focused on the requirements for practical symbol grounding, and review papers are cited where relevant.

2.1 Communicative symbols vs. physical symbols

There are differences between *physical symbols* and the *communicative symbols* described earlier by Peirce (Steels, 2008), differences that affect autonomous grounding, learning and usage. The studies in this thesis are concerned with learning *communicative* symbols, to allow mobile robots to communicate. However, many of the issues associated with communicative symbols have been presented as critiques of *physical symbols*. The following sections discuss communicative symbols, physical symbols and the differences between them.

Communicative symbols: Communicative symbols are symbols used for transferring information between multiple agents. The symbols refer to something by associations shared by all agents (Steels and Vogt, 1997). A communicative symbol must have a medium of transmission (e.g. light, sound, tactile), although the physical form of the symbol (the symbol represented within its medium(s)) does not need to have any resemblance to whatever is associated with the symbol (Hutchins and Hazlehurst, 1995). Communicative symbols differ from physical symbols in that they must be socially shared and transmittable.

Physical symbols: *Physical symbol systems* are AI programs that are described by their building blocks – physical symbols – and sets of rules that manipulate the symbols (Newell and Simon, 1976). Symbols are arbitrary patterns that are linked to values and computation is performed by applying rules to the symbols. Classic examples of physical symbol systems are the cognitive architectures SOAR (Laird et al., 1987) and ACT-R (Anderson et al., 1997). Both systems use symbols to represent the state of cognition and rules to read and modify the state. The selection and application of rules for both these systems are derived from psychological studies into human memory and cognition and both systems are capable of replicating human behaviors on carefully chosen tasks.

Physical symbols differ from communicative symbols in that they are used within computation and they do not need to be socially shared or transmittable. Physical symbol systems are used to support the *physical symbol systems hypothesis*:

> "A physical symbol system has the necessary and sufficient means for general intelligent action." (Newell and Simon, 1976)

There has been considerable debate about whether physical symbol systems are necessary for intelligence (Brooks, 1990), whether all computational systems are actually physical symbol systems (Vera and Simon, 1993), whether the human brain is based on physical symbols (Simon, 1990) and whether general intelligence is possible if physical symbols are not linked back to the world (Searle, 1980). Of these topics, this thesis is only concerned with the last, commonly known as the *symbol grounding problem,* which applies equally to both communicative symbols and physical symbols. The original ideas behind the symbol grounding problem were presented as critiques on physical symbol systems.

2.2 The symbol grounding problem

The symbol grounding problem famously arose in Searle's Chinese room thought experiment, which questions whether a mind can speak Chinese purely by manipulating input to produce output (Searle, 1980). Searle's Chinese room allows Chinese text to be transferred in and out, but is otherwise detached from the outside world. Inside the Chinese room, an English speaker (Searle) takes the Chinese character input, and uses a set of rules in a book to transform the input characters to output characters. Searle asserts that to an outside observer, the Chinese room appears to speak Chinese even though the person inside the Chinese room does not understand Chinese. Searle's conclusions were to differentiate between the properties of *Strong AI* (an AI that understands Chinese) and *Weak AI* (an AI that simulates the understanding of Chinese), and show that the Chinese room could only be considered Weak AI.

It is clear that the computation performed in the Chinese room closely resembles the ideas within physical symbol systems. The characters exchanged within the Chinese room are *ungrounded* – they do not have any *meaning* associated with them by the Chinese room (or Searle, who is in the Chinese room), although the symbols do have meaning associated by native Chinese speakers (Searle, 1980). The symbol grounding problem refers to the requirement of associating meanings with symbols, and the problem that symbols that are defined by other ungrounded symbols can never be resolved (Harnad, 1990). The symbol grounding problem has been described as the *Chinese merry-go-round*, where symbols that are defined only in terms of other symbols leads to a never ending recurse.

For a mind to understand symbols, they must be grounded in an agent's perception (Harnad, 1990). Grounding may be direct (i.e. symbol $\rightarrow$ sensors), indirect (i.e. symbol $\rightarrow$ symbol $\rightarrow$ sensors) or through abstraction (i.e. symbol $\rightarrow$ abstraction $\rightarrow$ sensors). The implications for ungrounded physical symbol systems are that i) the systems do not have any understanding of the meaning of a symbol, and ii) the human experimenter ascribes the relevant meaning to the symbol.

The articles by Searle and Harnad were both critiques of physical symbol systems; however, the symbol grounding problem applies equally to communicative symbols (Steels and Vogt, 1997). While it is possible that Searle's Chinese room could work as a language model for some static version of a language (such as Apple's Siri (Aron, 2011)), learning new symbols (among other issues) is likely to be more difficult (or even impossible) for a system that does not have language linked with perception (Steels, 2008). The dynamic nature of language requires the constant updating of meaning and introduction of new terms through a learning process (Steels, 2006). Learning and updating symbols is considered crucial to the symbol grounding problem within language.

Peirce's semiotic triangle provides a theoretical framework for symbol grounding in which a symbol is linked to a meaning (Ogden and Richards, 1923; Peirce, 1974). A symbol (a communicative word that two or more agents share) is associated with a referent (a feature of the environment or cognition) typically through association with sensors, actuators or cognitive state. An agent creates an internal representation of a referent through a process called private grounding. Two or more agents can then agree on a symbol to describe their shared experience through a second process called social grounding. The combination of private grounding and then social grounding allows anything perceivable by more than one agent to be socially labeled (see Figure 2.1).

Many solutions to the symbol grounding problem have been expressed within Peirce's semiotic frame-

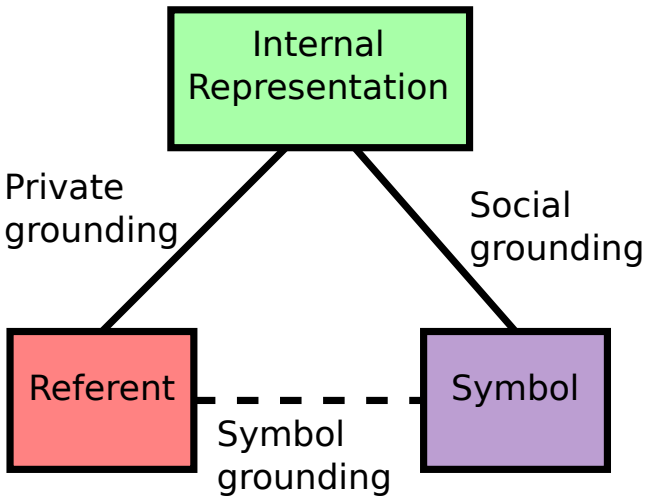


Figure 2.1: Peirce's semiotic triangle (adapted from Ogden and Richards (1923)). A referent, a feature of the environment is linked to a symbol, a communicative word through a two-step process. An agent first creates an internal representation for the referent in a private grounding step. Two or more agents then agree on a symbol to describe their corresponding internal representations in a social grounding step. The result is symbol grounding – the link between the referent and symbol.

work (Harnad, 1990; Barsalou, 1999; Sun, 2000). A solution to the symbol grounding problem was first suggested as a hybrid symbolic-connectionist system (Harnad, 1990). A physical symbol system is connected to an agent's perceptual representations by connectionist parts, thus linking the symbolic cognition of an agent to its environment.

A key question arising from this solution is *how are grounded symbols autonomously learned and linked to an agent's perception?* (Sun, 2000) A pair of agents that have pre-encoded symbols linked to perception are still limited by the inability to update and create new symbols (Steels, 2003). The ability to learn symbols and link them to meanings is a key aspect of communicative symbol grounding. Furthermore, language learning studies are particularly interested in socially coordinating and sharing symbols. Social learning (or at least social organization) of symbols is then essential (Steels, 2008). Social learning is typically implemented as some variant of a *conversation* (see section 2.2.1).

Grounding studies can be divided into two groups: cognitivism and enaction (Ziemke, 1999). Cognitivism is where primitive symbols in symbolic manipulation systems are linked to an agent's perception or representations as in the solution of Harnad (1990). Enaction is where symbols may not be required at all, resulting in direct connections between an agent's sensors and actuators (Brooks, 1990, 1991). Enaction emphasizes the concepts of embodiment - having a physical body and being situated within a physical environment – and ties cognition to these concepts (Ziemke, 2003). Some degree of embodiment or simulated embodiment is a prerequisite for autonomous symbol grounding, as a *body* that includes sensors and actuators is required for generating the perceptual input that forms the meaning of symbols.

Several studies have looked at grounding communicative symbols with embodied agents or simulated embodied agents to name shapes on a white-board (Steels, 2015) (see Section 2.4.1 below), other robots (Steels and Vogt, 1997), objects (Steels and Kaplan, 2002), colors (Steels and Belpaeme, 2005), actions (Tikhanoff et al., 2011) and spatial prepositions (Roy, 2002a; Steels, 2015) (see Coradeschi et al. (2013) for a recent review).

The following processes are common amongst all the embodied symbol grounding studies:

- conversations or other synchronization (see section 2.2.1),
- shared attention (see section 2.2.2),
- categorization (see section 2.2.3), and
- creating appropriate representations (see section 2.2.4).

These processes are described in the following sections.

2.2.1 Conversations for symbol grounding

Conversations, based on Wittgenstein et al. (1958), have been demonstrated as part of a solution for grounding symbols (Steels, 1995; Steels and Vogt, 1997). Conversations are a process undertaken by two agents that facilitate synchronization between the two agents and allow them to i) establish *shared attention* (see Section 2.2.2) and ii) transfer symbols. Conversations provide the social grounding process of symbol grounding (Schulz et al., 2011a).

Conversations are typically innate and can be as simple as communicating a single symbol that is linked to shared attention (Jung and Zelinsky, 2000). Additional words can be used to specify an object within a context (i.e. the word "where" designates talking about a place or spatially located object) (Schulz et al., 2011a) or extra-linguistic methods such as pointing or gaze direction can be used instead (Steels, 2015). Conversations have been implemented with feedback (Steels, 2015) and without (Baronchelli et al., 2006; Schulz et al., 2011a). Feedback allows agents to agree on lexical terms faster Steels (2015), but is biologically implausible (Bloom, 2002) and does not affect steady state coherence (Fontanari and Cangelosi, 2011).

A conversation typically consists of a question and answer (Schulz et al., 2011a; Steels, 2015). The question predicate provides the context for isolating a feature of an experience. Answering the question requires either creating or generalizing a symbol in order to describe the isolated feature. The symbols provided in a conversation are linked to aspects of the context.

2.2.2 Shared attention for symbol grounding

Shared attention is aligning two or more agents' attention onto the same feature or property of the environment. Shared attention is required for symbol grounding so that communicating agents agree on the same symbols for the same features (Steels and Vogt, 1997). Typically part of a conversation is used to synchronize the agents and to share attention through linguistic or extra linguistic means as described above.

A shared experience is a method of establishing attention on the same feature or property. A shared experience requires both agents to have the same experience, allowing them to ground symbols in referents formulated from those experiences. The key challenge for grounding through shared experiences is for both robots to attend to the same feature or property of the experience. Shared experiences can be established by looking at the same scene (Steels, 2015) or through imitation (Billard and Hayes, 1997), or by an evaluation of how close two agents are using markers (Vogt, 2002), audio transmission distance (Schulz et al., 2011a) or using some other proximity detector such as overhead tracking.

2.2.3 Categorization

A third process common to previous symbol grounding studies is the use of categorization. Categorization is the way features in the environment are arranged into categories that can be labeled, and is typically analogous to clustering. Typically for embodied agents, the number of different perceptual states is very large (e.g. consider a 320×240 RGB24 image - the number of possible perceptual states would be $2^{24\times320\times240} \approx 3.076 \times 10^{554860}$). Instead of assigning a name to every individual state, the similarity between perceptual states is exploited to reduce the number of options.

Categorization can be discriminative or generative (Roy, 2002b), with discriminative providing coverage of the entire feature space, but generative providing more flexibility (see Figure 2.2). A variety of methods are used for categorization including discrimination trees (Steels, 1999), clustering through neural networks (Hutchins and Hazlehurst, 1995; Tikhanoff et al., 2011), neural field modeling (Fontanari and Perlovsky, 2008), Gaussians (Roy, 2002a) or kernel density estimators (KDE) (Schulz et al., 2011a).

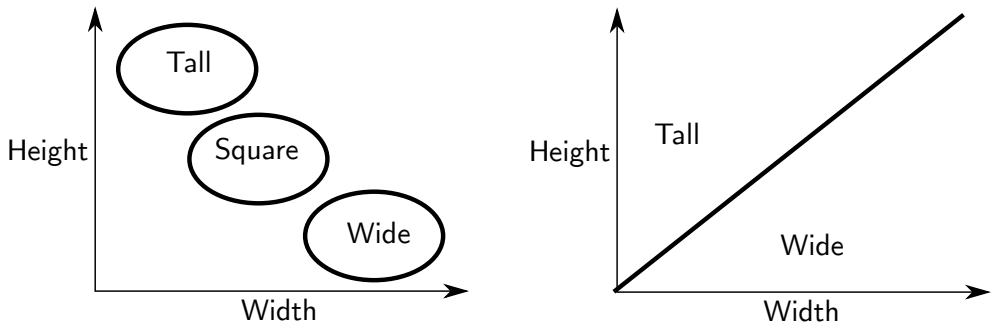


Figure 2.2: Generative vs discriminative models (adapted from Roy (2002b)). Model a) is generative and model b) is discriminative.

Each of these methods have advantages and disadvantages. Discrimination trees allow discrimination thresholds to be learned by an agent so that it can use perceptual features to differentiate between different objects. This allows the requirements of the conversation to dictate the way an agent divides perceptual features. Discrimination trees are only able to discriminate by using thresholds for each feature, i.e. a function of multiple features cannot be used.

Neural networks are used in two different ways. Neural networks can be used as *auto-associators* (see Kramer (1991)) and the compressed representations used as the categories (Hutchins and Hazlehurst, 1995). Neural networks can also be used as classifiers, with a language output that corresponds to categories (Tikhanoff et al., 2011). In both cases, the categories created can span multiple dimensions. The advantage of the auto-associator is that learning is unsupervised and it becomes akin to dimensionality reduction, with a number of dimensions specified as a prior parameter. This is particularly suited to bootstrapping language. Neural network classifiers typically use supervised learning and for bootstrapping require other means for creating new categories.

Neural field modeling uses Gaussians to create clusters based on a set of partial differential equations expressing the likelihood of the Gaussians capturing the dynamics of a category (Perlovsky, 2001). The

equations are solved iteratively, which changes the mean and variance of the Gaussians to best capture categories. The categories represented by neural field modeling are also able to span multiple dimensions.

Gaussians are also used to form categories with standard statistical estimators (Roy, 2002a). In Roy's DESCRIBER architecture, categories are modeled using the mean and variance parameters extracted from human descriptions of a scene (see Section 2.4.3).

2.2.4 Representations required for symbol grounding

Symbol grounding requires agents to store the links between words and meanings. The majority of studies use either neural networks (Hutchins and Hazlehurst, 1995; Tikhanoff et al., 2011), a variant of the lexicon table (Steels, 1999; Roy, 2002a; Schulz et al., 2011a; Tellex et al., 2011) or a parameterized mapping (Fontanari and Perlovsky, 2008).

Neural networks are able to store links between words and meanings within their weights (for details of how neural networks work see Hornik et al. (1989)). The weights can be learned through the standard back-propagation algorithms (see Rumelhart et al. (1988)) by providing words and meanings on the input (Tikhanoff et al., 2011) or by using the neural network as an auto-associator and using the compressed representation in the hidden layer to form symbols (Hutchins and Hazlehurst, 1995).

Neural networks are capable of generalizing words just by providing different input patterns; however, they are also slow to learn and may require many samples of words and meanings before converging. The slower learning rates mean that they are not capable of one-shot learning, which is often desirable for language learning.

Lexicon tables are more flexible than neural networks, because they store only links between words and meanings as a directed graph. Words can be linked to categories (Steels, 2015), or they can be linked to the instances directly (Schulz et al., 2011a). Lexicon tables allow one-shot learning, as links can be created immediately during a conversation. Lexicon tables are particularly flexible in the choice of categorization used, since unlike neural networks and parametrized mappings, they do not enforce a particular clustering algorithm. However, this flexibility can also be a disadvantage, as the categorization used with lexicon tables is only as good as the clustering algorithm chosen.

Parametrized mappings describe the case where words are treated indifferently from features, and are considered as just another input to the categorization algorithm (Fontanari and Perlovsky, 2008). In this case, the information about a link between a word and meaning is contained within the resulting categories. Parametrized mappings of this sort allow for the model to generalize across words as if they were features. This can allow words to be temporal signals, or contain noise.

2.2.5 Measuring success in symbol grounding

It is important to be able to measure success in symbol grounding studies. Several measures of success have been used in symbol grounding studies: language game success (Steels, 2015), task performance (Jung and Zelinsky, 2000; Schulz, 2008), human comprehension (Roy, 2002a), communicative success (Kirby and Hurford, 2002) and coherence (Schulz et al., 2011a).

The metric of language game success relies on language games having a measurable goal for two robots playing the game; i.e. the ability to identify a shape (Steels, 2015). In this case, two robots hold a language game in which learning is coupled to the reinforcement or supervised feedback given.

Task performance relies on having a separate task to test the performance of language. Tasks need to be cooperative, but can either be many trials (i.e. going to a given location (Schulz et al., 2011a)) or a single task that requires repeated communication (i.e. cleaning an area (Jung and Zelinsky, 2000)).

Human comprehension and communicative success are only applicable to specific cases. Human comprehension is using humans to evaluate the speech output from an agent and is a good measure for agents that are able to output an understandable level of natural language (Roy, 2002a). Communicative success involves directly comparing the meanings produced by agents when they decode a word. It is only applicable to agents that are presented with identical meanings, such that they can be shown to decode a symbol to the same values.

Coherence is similar to communicative success, but compares two agents' entire lexicons over continuous features, and relies on using the environment as a guide for the comparison of meanings (Schulz et al., 2011a). A grid is imposed on the continuous features, and the label given to each grid square is compared between two agents. Coherence is a good measure when features are continuous and when the words an agent produces are considered to be as important as comprehension. That is, it is a good measure when it is important that the agents produce the same words for the same features (the opposite is that the agents understand each other but use different words).

2.3 Beyond solving the symbol grounding problem

In 2008, Steels claimed that the symbol grounding problem had been solved, suggesting that with embodiment, language games, clustering and private and social symbol organization, agents are capable of symbol grounding (Steels, 2008). However, Steels' solution has some limitations for application.

Steels' solution applies to grounding perceptual referents (i.e. features of the environment that are directly perceptible), whereas concepts such as space and time can not be grounded directly in perception and must instead be indirectly grounded in internal representations (Schulz et al., 2011b). Grounding in different sensors and cognition is closely related (Jung and Zelinsky, 2000) and invalidates some of Steels' assumptions about agents' abilities to discriminate. Another open problem is how agents can autonomously ground symbols in other symbols (Cangelosi, 2011). The following sections look at several of these still open problems:

- grounding symbols in other symbols (see Section 2.3.1),
- dealing with referential uncertainty, a problem closely associated with symbol grounding (see Section 2.3.2),
- grounding in cognitive processes and states that are only indirectly linked to perception (see Section 2.3.3 for grounding of space and time), and
- grounding symbols across different sensors and cognition (see section 2.3.8).

2.3.1 Grounding symbols in other symbols

Solutions to the symbol grounding problem require that symbols be eventually grounded in sensors and cognition. Symbols can be grounded in other symbols, so long as "terminal" symbols are eventually grounded in sensors. Higher-level symbols can be grounded using combinations of entry-level symbols, in a process called *grounding transfer* (Cangelosi et al., 2000). Entry-level symbols are first grounded directly in perception, and then higher-level symbols are created by categorizing across entry-level symbols. Grounding transfer has also been extended to forming object definitions from the symbols that make up the parts of the object (Riga et al., 2004) and to creating distances and directions from locations (Schulz et al., 2012). In the latter case, distances were defined by two locations and directions were defined as the angle between two locations from the viewpoint of a third location. New locations could also be defined by a known location, a distance and a direction in a process called *generative grounding*, allowing an agent to ground a location that is not actually perceived by the agent. In this way, agents can extend shared attention to objects and places that they can not perceive.

Another extension of grounding transfer is *symbol attachment*, where, like grounding transfer, symbols are defined by other symbols, but additionally a symbol's meaning is constantly dependent on the symbols used to define it (Sloman and Chappell, 2005). Symbol attachment is the most flexible representation for grounding symbols in other symbols; however, complex graphs of symbols can be created (such as cyclic graphs) that may be difficult to work with.

2.3.2 The referential uncertainty problem

Autonomous symbol grounding requires dealing with referential uncertainty. Referential uncertainty refers to the inability of an agent to unambiguously associate a word with its meaning due to multiple candidate associations in the agent's context. For example, if a native pointed at a rabbit and said "gavagai", someone listening would not know if the native was referring to the rabbit, parts of the rabbit or even something unrelated to the rabbit (Quine, 1960). An infinite number of meanings could be associated with any given context.

Referential uncertainty is closely linked to an agent's ability to share attention. Attentional mechanisms, such as only learning unambiguous words (i.e. words to which shared attention is identical and to a single referent) (Jung and Zelinsky, 2000), using additional language to specify attention more precisely (Schulz et al., 2011a) and cross-situational learning (XSL) each have different effects on shared attention that can be used to mitigate referential uncertainty.

Several studies look purely at XSL. These studies typically define an abstract set of meanings $M = \{M_1, M_2, ..., M_n\}$ that relate to a set of words $W = \{W_1, W_2, ..., W_n\}$. A context is then defined as a word and an associated set of meanings $C = \{W_1, M_1, M_2, ..., M_{cn}\}$. Agents are presented with many contexts, and rules or statistics are used to infer the correct words and meanings. This inference can use a simple set of rules (Siskind, 1996) or statistics (Smith et al., 2006) to identify the intended meaning in a set of meanings from a set of words. Statistical approaches use the counts of words, meanings, and the conditional *meanings given words,* to identify the association between the correct word and meaning, and the number of context presentations required to learn this association (Smith et al., 2006; Smith and Yu,

2008; Blythe et al., 2010).

Studies using abstract meanings assume that categorization is performed before naming. If categorization is performed when naming, the referential uncertainty problem is not solvable in a closed form. Studies have shown how mutual information (Oates, 2003), Kullback-Liebler (KL) divergence (Roy, 2002a) or neural modeling fields (Fontanari et al., 2009) can be used to generate the statistics of word and meaning usage in these cases.

2.3.3 Grounding space, time and other non-perceptual referents

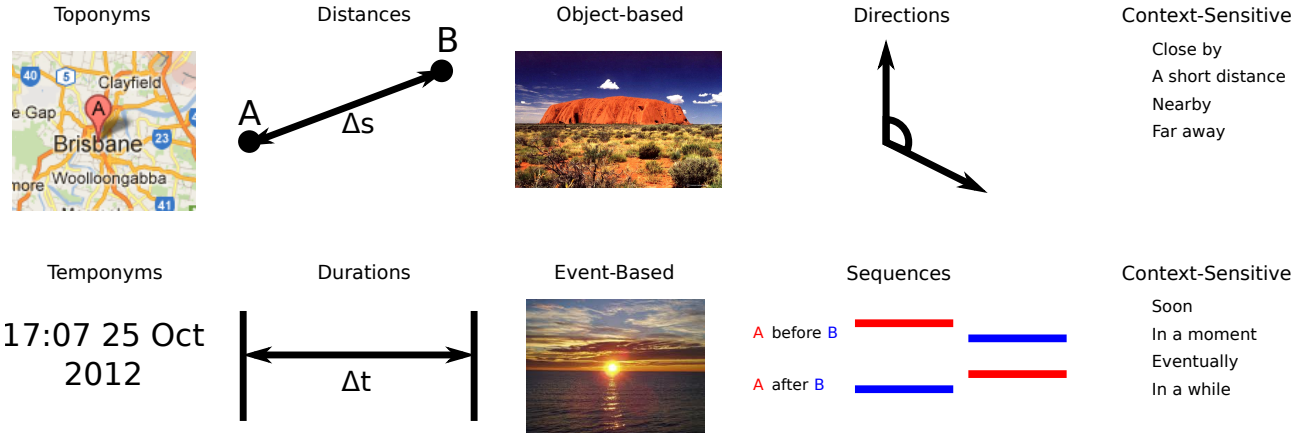


Figure 2.3: Some (but not all) of the different types of space and time used in natural language, aligned on their metaphorical similarities. Toponyms and temponyms respectively refer to a single point in space and time, distances and durations refer to a range in space and time, object-based and event-based refer to space and time relative to objects and events, and context-sensitive terms apply to both space and time. Directions and sequences do not metaphorically align easily across the space-time boundary.

Space and time are foundational in human and robot cognition and although they have strict physical definitions (Feynman, 1948), expressions in natural language have formed around how space and time are indirectly perceived by humans as sequences of perceptions (Gibson, 1975; Frank, 1992; Engberg-Pedersen, 1999; Kuipers, 2008) (see Figure 2.3). Different cultures understand and express space and time through different metaphors and events (Levinson, 1996; Núñez and Sweetser, 2006; Evans, 2010; Sinha et al., 2011). The indirect perception of space and time also applies to embodied robots. Terms for space and time can not be grounded directly in perception, but instead must be grounded in cognitive processes. The importance of space and time is acknowledged by the large amount of research into robot navigation and localization in recent years (Grisetti et al., 2007; Montemerlo and Thrun, 2007a; Davison et al., 2007; Milford and Wyeth, 2010; Maddern et al., 2012).

2.3.4 Spatial language learning

Grounding studies have looked at developing spatial relations (Steels, 1995, 1999; Roy, 2002a), route descriptions (Kollar et al., 2010; Tellex et al., 2011) and locations (Jung and Zelinsky, 2000; Schulz et al., 2011a). Semantic spatial relations and object-based space can be learned from static scenes; however, grounding locations requires access to more complicated spatial representations (Schulz et al., 2011a). Creating more complicated spatial representations usually requires simultaneous localization and mapping.

2.3.5 Simultaneous localization and mapping

In the robotics literature, a robot's representation of space depends on the robot's representation of its location and environment. Often, information about the absolute position of a robot (i.e. Global Positioning Satellite (GPS) information) is not available to a robot or is not accurate enough for a robot to rely on. In these cases, in order for a robot to represent its environment, it must both create a map of the environment, and localize itself within that map. The joint dependencies of creating a spatial map, and localizing within that map require joint evaluation in a process termed Simultaneous Localization and Mapping (SLAM) (Montemerlo and Thrun, 2007b).

The SLAM problem is often described probabilistically as calculating:

$$P(x_k, M|Z, U, x_0), \tag{2.1}$$

where x_k is the robot's pose at time step k, M is a vector of landmarks, Z is a matrix of individual observations z_{ik}, taken of landmark i at time step k, U is the vector of control inputs to the robot, and x_0 is the robot's initial pose (Durrant-Whyte and Bailey, 2006). Algorithms often divide the solution into two parts: a distinct time update and measurement update. The time update predicts a new state from the previous one using:

$$P(x_k, M|Z^{k-1}U, x_0) = \int P(x_k|x_{k-1}, u_k) \times P(x_{k-1}, m|Z^{k-1}, U^{k-1}, x_0)dx_{k-1}, \tag{2.2}$$

where Z^{k-1} is the matrix of observations up to $k-1$ and U^{k-1} is the vector of control inputs up to $k-1$. A measurement update corrects the state using the current observations (z_k):

$$P(x_k, M|Z, U, x_0) = \frac{P(z_k|x_k, M)P(x_k, M|Z^{k-1}, U, x_0)}{P(z_k|Z^{k-1}, U)}. \tag{2.3}$$

Since the inception of SLAM almost three decades ago (Cheeseman et al., 1987), there have been many systems implemented within the framework described by Equations 2.1-2.3 (Ho et al., 2015). Typically SLAM systems are implemented as *filters*, algorithms that receive noisy observations and produce the "cleaner" location of the robot and poses. Common groups of SLAM systems are those based around variants of the Kalman filter (Leonard and Durrant-Whyte, 1991; Huang et al., 2009) and particle filter (Montemerlo et al., 2002; Grisetti et al., 2007; Montemerlo and Thrun, 2007a). Kalman filter variants provide Bayes-optimal methods for updating based on predictions and observations; however, they suffer from i) a reliance on Gaussian distributions to represent sources of noise, and ii) quadratic computational complexity (Montemerlo et al., 2002). Particle filter implementations are non-parametric, and allow the representation of arbitrary distributions, which addresses the the first of the Kalman filter problems. Implementations using particle filters have also addressed computational complexity through innovative decomposition of SLAM into separate steps for localization and then landmark estimation conditioned on location (Montemerlo et al., 2002).

SLAM systems employ a variety of sensors for observations. Different systems have used cameras (Davison et al., 2007; Milford and Wyeth, 2010; Maddern et al., 2012), laser scanners (Montemerlo et al., 2002; Grisetti et al., 2007; Montemerlo and Thrun, 2007a), lidar (Kohlbrecher et al., 2011), Kinects

(Engelhard et al., 2011), microphones (Munguía and Grau, 2008), or WiFi signals (Ferris et al., 2007) for observing landmarks. Landmarks can be exact matches to sensor data (Grisetti et al., 2007), features extracted from the sensors (Davison et al., 2007; Maddern et al., 2012), or down-sampled sensor data (Milford and Wyeth, 2010; Milford and Wyeth, 2012) Additionally, wheel odometry is commonly used in place of the control inputs within the time update step (Equation 2.2), as the readings are typically more accurate than the control inputs.

There are a variety of ways to represent maps within SLAM systems, including, locations of landmarks, topological graph and occupancy grids. Representing locations of landmarks relies on storing a pose for every landmark known by a system. This method is typically implemented with Kalman filters (Bailey et al., 2006) and is constrained to small numbers of features (Montemerlo et al., 2002). Occupancy grids represent space as a grid of values where different values indicate obstacles, free space and unknown areas (Thrun, 2003). By representing a map by using a grid, the computational complexity is effectively a function of the resolution of the grid, and not the number of landmarks that can be represented. This has the limitation of imposing a fixed resolution over the environment; however, variants of the occupancy grid additionally allow adaptive resolutions (Montemerlo and Thrun, 2004). Topological representations consist of a graph where the edges represent the trajectory of an agent, and the nodes represent points of interest and branch points. Topological representations mitigate computational complexity by sampling at intervals along a trajectory. The graph structure of topological maps allows semi-metric representations of the world, where distances, rotations and scales can be linked to the robot's sensors instead of exactly to the environment (Milford and Wyeth, 2010).

Two state-of-the-art SLAM systems are used in the studies in this thesis - RatSLAM and Gmapping. Background is given on both in the following sections.

RatSLAM: RatSLAM is a biologically inspired system based on research into the rat hippocampus (Milford and Wyeth, 2010). RatSLAM does not exactly fit into either of the two groups of SLAM systems described above. RatSLAM uses a continuous attractor network to represent and filter the uncertainty about the robot's location within its environment similarly to the representations of particle filters. However, unlike particle filters, the continuous attractor network represents the uncertainty of the robots pose with uniform samples within a 3D cube that represents a 2D square in the environment, and all possible rotations in the third dimension. The practical RatSLAM implementation consists of three modules: i) a visual template matching module, ii) a pose cell network and iii) an experience map. The visual template matching module receives images from the agent and attempts to match them with previously presented images and the associated location. The pose cell network provides the real novelty in RatSLAM as it integrates location information from the visual template matching module with odometry information from encoders and produces a relative estimate of location. The setup of the pose cell network is similar to a grid cell in a rodent in that it "fires" in grid patterns as the robot moves through space (Milford et al., 2010). The experience map receives the output from the pose cell network and also the raw odometry from the robot, allowing it to create and then refine a topological map. The experience map is not explicitly based on a rodent's hippocampus, but the way experiences are linked together is similar to an episodic memory (Tulving, 1983). In addition, after the robot has visited a location from multiple directions and corrected the map, the nodes in the map take on similar properties to that of place cells in the hippocampus

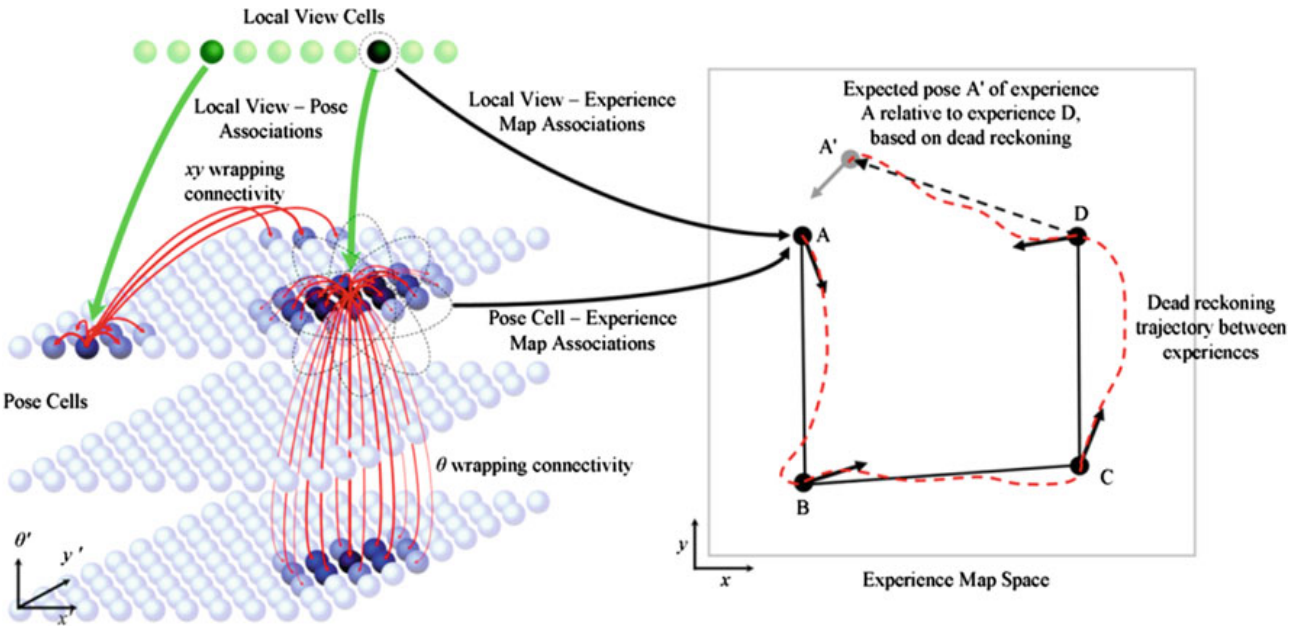


Figure 2.4: The RatSLAM architecture (reproduced from Milford and Wyeth (2010) with permission). A robot's current vision (as an image) is matched to stored image templates, with each template activating a different local view cell (top left). The local view cells are linked to small numbers of pose cells. Activating a local view cell activates its associated pose cells. The pose cells are organized with Continuous Attractor Network (CAN) dynamics, causing activity to settle into a single region. The robots motion causes activity in the pose cells to move in the direction of motion of the robot. Activity packets wrap at the edges of the three dimensions. Both local view cells and pose cells are linked to small numbers of experience nodes, within a semi-metric map called the experience map.

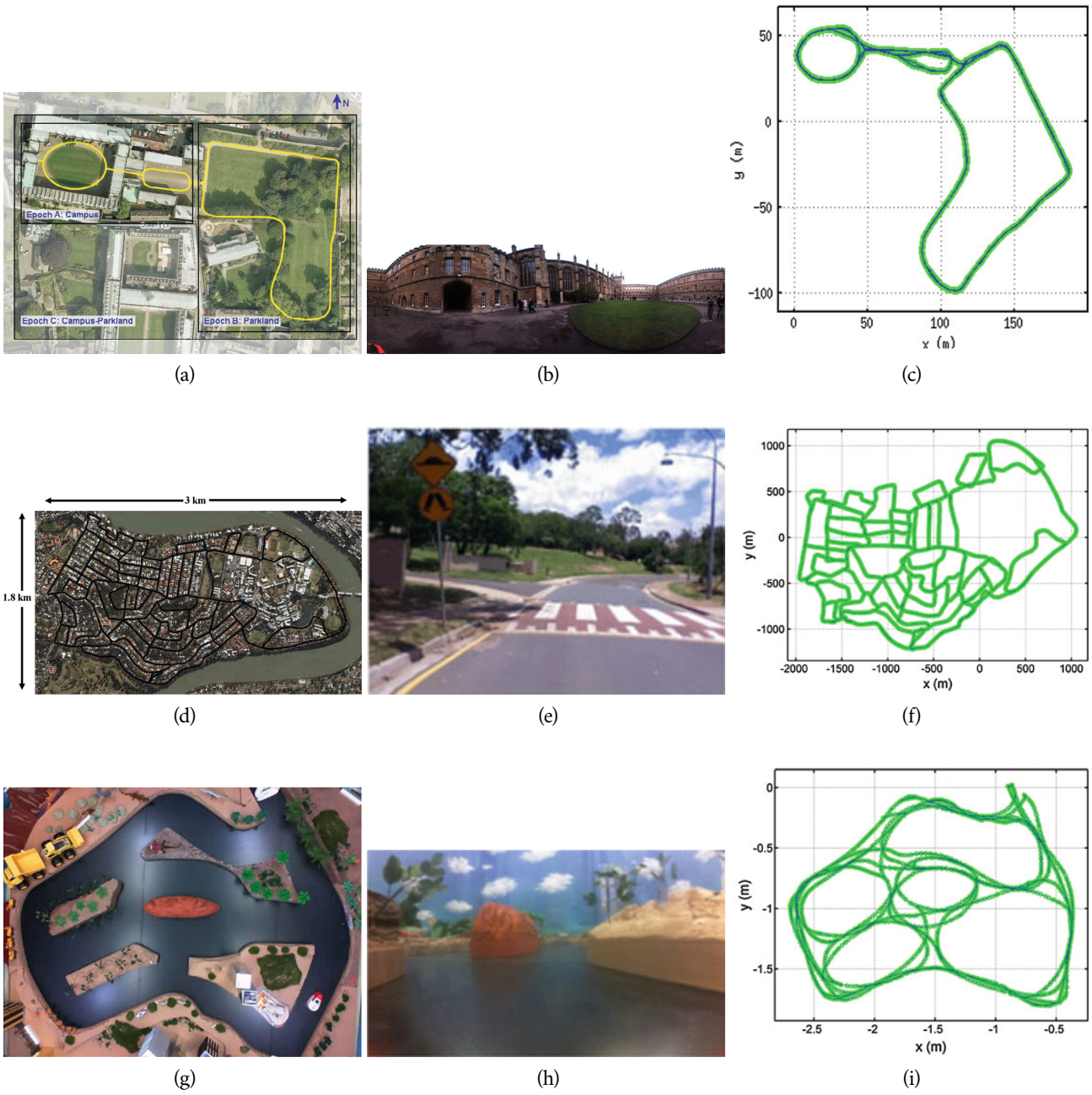
(O'Keefe and Nadel, 1978).

RatSLAM has been tested using a variety of platforms and environments (see Figure 2.5). A detailed description of the OpenRatSLAM implementation is provided in Ball et al. (2013), which is included in this thesis as an appendix (see Appendix A).

Gmapping: Gmapping is an explicitly probabilistic approach, based on particle filters (Grisetti et al., 2007). Each particle in Gmapping represents a different version of the entire environment as an occupancy grid, with the computational complexity managed by maintaining a low particle count. In particular, the dependency between the robot's map and trajectory is exploited to reduce the number of particles required to obtain a good estimate of the state of the system - an optimization called Rao-Blackwellization (Doucet et al., 2000).

The reference implementation of Gmapping produces static maps that are not easily modifiable. It is instead typical to first create a map using Gmapping and then use Adaptive Monte-Carlo Localization (AMCL) (Fox et al., 1999) to localize using the map. AMCL is based on particle filters where each particle represents a possible location of the agent. AMCL does not modify the occupancy grid provided by Gmapping.

Grounding language in SLAM representations: Several studies have looked at grounding language in SLAM systems using self-sorting maps (Jung and Zelinsky, 2000), RatSLAM maps (Schulz et al., 2011a) and in Gmapping maps (Tellex et al., 2011). These studies are described in more detail in section 2.4.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 2.5: The results of RatSLAM mapping several datasets (reproduced from Ball et al. (2013)) of the Oxford New College dataset (see Smith et al. (2009)); a) the robot's journey, b) an image from the journey, c) the resulting map; the mapping of St Lucia (see Milford and Wyeth (2008)); d) the robot's journey, e) an image from the journey, f) the resulting map and the mapping of the iRat's Australia maze (see Ball et al. (2013)); g) the environment, h) an image from the iRat within the maze, i) the resulting map.

Two of the studies used shared attention mechanisms that allowed both robots to refer to the same location in their cognitive maps during a conversation (Jung and Zelinsky, 2000; Schulz et al., 2011a). The robots were then able to create shared symbols for these locations and use them again later for different tasks. The last study focused on routes and landmarks present within a semantic map, using a variety of data sources (Tellex et al., 2011). The robots learned the landmarks and routes from generated datasets.

2.3.6 Robot path planning and navigation

For robots to have full spatial cognition, they require more than just the ability to represent space - robots also need to be able to use their representations of space to perform tasks. One of the most common spatial tasks that mobile robots need to be able to do is to plan paths and navigate to goals within their environments. Although it is possible for robots to navigate without maps (e.g. using environment cues only), using a map enables a robot to maintain learned knowledge about the environment. In order to plan a path, a robot requires a map, a representation of the robot's location within the map, and a goal location (Meyer and Filliat, 2003).

Different types of maps affect the types of path planning that are applicable. Topological maps are graphs of the environment, so Dijkstra's algorithm can be directly applied (Dijkstra, 1959), or more efficient variants such as A* (Hart et al., 1968). However, issues that require extended solutions for topological maps are taking shortcuts, and adapting to dynamic blockages. Occupancy grids can also be thought of as a graph, where each square is linked to the 8 squares surrounding it. Dijkstra's algorithm and A* can also be applied to this case. An advantage of occupancy grids here are that both taking shortcuts and adapting to dynamic blockages become possible (Stachniss et al., 2004).

The more difficult case for path planning is when representing a map as a set of landmarks (Meyer and Filliat, 2003). In this case the landmarks and space around the landmarks must first be converted to an occupancy grid, topological map or some other discretized representation often referred to as a robot's configuration space (C-space) (Sariff and Buniyamin, 2006). There are many ways of creating the C-space by dividing space into convex polygons (Latombe, 2012), rectangles (Arleo and Gerstner, 2000), Voronoi diagrams (Bhattacharya and Gavrilova, 2008), cones (Brooks, 1982) or quad trees (Kambhampati and Davis, 1986). Once the C-space is formed, algorithms such as A* and Dijkstra's algorithm can again be applied.

The output of path planning is a navigation strategy that fits into one of two groups (Meyer and Filliat, 2003): i) a set of moves to be sequentially executed, or ii) a set of reactive behaviors that are constantly run that cause goal-directed movement (Schoppers, 1987). The second group is better equipped to deal with dynamically changing environments.

Within the RatSLAM and Gmapping SLAM systems, the path planning algorithms are variants of Dijkstra's algorithm that is used with the topological map of RatSLAM (Heath et al., 2011) or with the occupancy grid of Gmapping. The navigation system in RatSLAM fits into the *reactive behaviors* group and is described further within the methodology chapter (see Section 3.1.1) (Heath et al., 2011). For navigating using maps from RatSLAM systems, a robot is required to localize within the map while following the planned path.

2.3.7 Temporal language learning

Time is often understood and expressed through spatial metaphor (Clark, 1973; Lakoff and Johnson, 1980; Boroditsky, 2000; Gentner, 2001; Moore, 2006) although there are limits to this mapping - in particular transience is attributed only to time (Galton, 2011). Little has been studied on temporal cognition for robots (Maniadakis and Trahanias, 2011), although SLAM representations of episodic memories can be seen as a form of temporal cognition (Schulz et al., 2011b). Robots similar to those in the Talking Heads

project (see Section 2.4.1) were able to ground event descriptions, and the representations include the order of events and the interval of events (Steels and Baillie, 2003). The event descriptions hint at the grounding of temporal relations, but no results are given. In another study, a simulated ontology of time was formed based on the evolution of language approach and discrimination games of Steels (De Beule, 2006). It was assumed that the agents had access to high-level representations of events as predicates, such as $fall(X)$ and $past(X)$, and could form sequencing of past and present when required. Previous Lingodroids studies showed how temporal terms could be grounded in both shared journeys and RatSLAM routes (Schulz et al., 2011b).

Within temporal language learning, there are many different types of time that are used in natural language, but have not yet been studied on robots (see Figure 2.3). Some of these, such as event-based time are used as the only way of expressing time by some remote cultures (Sinha et al., 2011), and can be practically useful for specifying co-occurring events. Robots would benefit from more advanced temporal cognition in a variety of activities (Maniadakis and Trahanias, 2011).

2.3.8 Grounding across different sensors and cognition

Heterogeneous robot teams are a growing research area, involving robots with a variety of abilities interacting, cooperating, coordinating and communicating. Examples of tasks for teams of robots include environment mapping (Simmons et al., 2000), cooperative localization (Parker et al., 2004), search and rescue (Murphy et al., 2000) and decentralized environment modeling (Gil Jones et al., 2006). A key challenge for robots that are part of heterogeneous teams of robots and humans is how to communicate about information in their respective knowledge bases, formed through their individual interactions with the world. The shared language used for communication must be grounded in each robot's own representations.

The only study that has investigated symbol grounding across different sensors is that of Jung and Zelinsky (2000), which is described in more detail in section 2.4.2. The robots in this study were able to ground terms in cognitive maps; however, the robots differed in sensors only, as their cognitive maps were identical.

2.4 Language learning models

Frameworks that integrate different facets of language learning together provide research tools for exploring the origins of language (Steels, 1999; Schulz et al., 2011a), human-robot interactions (Roy, 2002a; Tellex et al., 2011) and general artificial intelligence. Language learning frameworks are important because they combine theories about language learning with practical systems that are capable of implementing parts of these theories. The practical parts of the framework help to ensure that the theories are self-consistent. Language learning frameworks complement experimental research into human-robot interaction and natural language by providing context for experimentally observed phenomena. When studying robot-robot communication, language learning frameworks can identify requirements and sufficiencies of communication between robots and provide insight into the nature of semiotics, semantics and grammar.

Language learning frameworks need processes to handle the groundings described by Peirce's semiotic triangle, creating internal representations of the environment and socially associating a symbol with an internal representation (Ogden and Richards, 1923). Frameworks that can produce language need to support invention of terms, recall of a term to describe an internal representation and usually (but not always) generalization of terms to additional private representations. Frameworks are developed for different types of agents (embodied robots, simulated robots and software agents), different numbers of agents, for online and offline usage and for use with other agents or humans or both.

As described in the Introduction to this thesis (Chapter 1), there are no language learning frameworks that adequately address all the requirements for language learning on mobile robots. However, there are many language learning frameworks that handle some of these requirements. The following sections review these frameworks and studies. The language learning frameworks are chosen and analyzed based on the following abilities:

- ability to ground in cognition (indirectly grounding in perception),
- ability to ground across sensory and cognitive differences,
- ability to deal with referential uncertainty,
- ability to learn after one epoch (one-shot learning of new words),
- ability to learn online (learn at the same time as maintaining a usable lexicon),
- ability to produce words, and
- ability to generalize words to new features.

2.4.1 Talking Heads

In robot-robot language interactions, the Talking Heads project is arguably still one of the state-of-the-art frameworks (Steels, 2015). The pairs of agents held almost 500,000 interactions and were able to create a stable vocabulary of 300 words. The Talking Heads project used cameras as the embodiment for pairs of robots. A population of software agents were able to inhabit the "talking heads". The two cameras were positioned near each other and pointed towards the same scene – a white-board with shapes on it. The robots were able to learn to describe different shapes on the white-board by discriminating across an innate set of dimensions.

Private grounding: The Talking Heads used a structure called a discrimination tree for private grounding. Discrimination trees allow white-board shapes to be described by ranges of values for each dimension, contained within a decision-tree-like structure. Dimensions could be dynamically subdivided into ranges to discriminate between different shapes. The use of discrimination trees allowed the Talking Heads to create categories that directly correspond to the minimal discriminating dimensions of a shape.

Social grounding: The Talking Heads project pioneered the use of the language games of Wittgenstein et al. (1958), using highly-structured, innate grammars to allow agents to synchronize and indicate shared experiences. The major language game in Talking Heads is called the guessing game and it consists of the following steps.

1. One agent (the speaker) selects a context (a set of shapes) and a shape (the topic). Both agents are aware of the context, but only the speaker knows the topic.
2. The speaker chooses a set of features that discriminate the topic from the other shapes in the context.
3. The speaker looks up the best word it has for the set of features and communicates that to the listener.
4. The listener looks up the communicated word and finds the shape that best matches.
5. The listener indicates the shape to the speaker.
6. The speaker provides feedback about whether the listener is correct.
7. If the listener is incorrect, the speaker provides further feedback until they both agree on a topic.

Feedback within language games is controversial. As noted previously feedback is not present in early infant language learning (Bloom, 2002), it is possible to learn a lexicon through language games without feedback (Schulz et al., 2011a) and Fontanari and Cangelosi (2011) demonstrated that gains are mainly short term.

The Talking Heads maintain mappings between symbols, discrimination trees and confidence using lexicon tables. The lexicon tables are updated during the guessing games. After many games, the lexicons of a population eventually converge so that the same word is used to describe the same features.

Production and generalization: The Talking Heads produced words as part of social grounding by first deciding what was required to discriminate a shape from other shapes on the white-board, then choosing words to best describe the discriminating tree. If no such words exist, then words could be invented and associated with the discriminating ranges. The discrimination trees therefore also defined the possible generalization of a word. The ranges of values produce hyper-prisms as the generalization regions of words. The discrimination tree approach allows both top-down and bottom-up influences – finding words that discriminated the objects in a scene was influenced by both the sensor readings of the scene and also the lexicons that the robots had already obtained.

Characteristics: While the Talking Heads project is a well-known and successful project, it has limitations for application to space, time and different mobile robots. The scenes that the Talking Heads look at are simplistic and not indicative of real world data. The embodiment of robots as cameras on stands is limited in the sensors and actuators that the agents have access to (although other studies have implemented the guessing game on mobile robots (Steels and Vogt, 1997; Steels, 2001), but these studies also had limited sensors). The Talking Heads ground directly in perception, and never in cognition.

The robots handle referential uncertainty through their use of discrimination trees and corrective, non-linguistic feedback. Only naming discriminating features reduces the context in learning, and therefore reduces referential uncertainty. The robots pointing completely removes referential uncertainty. As described previously, pointing is limited to the spatial world – i.e. it is not possible to point at *time*.

2.4.2 Symbol grounding for heterogeneous cleaning robots

Jung and Zelinsky's robot study shows how two cleaning robots are able to use grounded communication to perform a vacuuming task more efficiently (Jung and Zelinsky, 2000). One robot has a camera and brush, while the other robot has only "whiskers" and a vacuum. The robots are able to ground locations as points in their topological maps so that the camera robot can then use these names as references to tell the other robot where dirt is. Jung and Zelinsky demonstrate that the performance of the robots improve when the grounded communication is enabled. The cleaning robot study is notable as it is the only grounding study that features robots with different sensors and actuators.

Private grounding: The two agents use SLAM based on self-organizing maps to create allocentric spatial maps. Nodes within the maps form the private groundings of the agents within this study.

Social grounding: Social grounding is an implicit conversation in the study. Both robots share attention to the location of the vacuuming robot (i.e. both robots share attention to the location of *one* robot) and the robots acknowledge that location. Both robots then internally associate the next numeric value in their identical sequences with that location.

Production and generalization: Production is simplistic within this study – the numeric labels are provided to refer to a place (to indicate dust). Distances in encoder counts can optionally be combined with a location label, although there are no labels or groundings for these distances. There is no explicit generalization in this study – a new label is used for every node referred to.

Characteristics: The novelty of this study is the different sensors used by the different robots, the groundings in cognitive maps and the practical task; other aspects of language learning have several limitations. Although the two robots used have different sensors, cognitively the two robots are identical, sharing the same map representations. There are limitations to the grounding of the robots - words only refer to a single point in the map and are not generalized. The robots deal with referential uncertainty with a simplistic method - they know *a priori* that they only communicate about space, which allows for one shot learning of terms for space but does not allow learning anything other than space.

2.4.3 DESCRIBER

The DESCRIBER architecture of Roy is notable for its requirement of creating language constructs (Roy, 2002a). Where most other human language learning studies were attempting to form models to comprehend aspects of human speech, the scene description task performed by DESCRIBER required an agent that could both develop a model of language and use the model to describe rectangles in a simple scene. In

order to achieve this DESCRIBER requires models of both the meanings of words and a representation of grammar. DESCRIBER was tested with human experimenters by describing a shape in the scene to the human and then the human selecting the shape that they think best matches. DESCRIBER demonstrates close to human performance at this task, with an average of 81.3% of agent-described shapes correctly selected by three humans, compared with 89.8% correct from human descriptions.

Private grounding: The agent projects values for a topic onto preset dimensions such as width, height, width/height ratio, color and position. The private grounding is formed from Gaussians that are used to categorize the values within dimensions. KL-divergence is used to greedily link multiple Gaussians to describe a shape.

Social grounding: When learning the model, the agent is presented with a number of contexts, which include indication of the topic - a rectangle in the scene and a description of the topic. Words are organized into classes based on both their associated meanings and the co-occurrence statistics. Words are initially associated with a Gaussian representation in each dimension. Only Gaussian representations with high KL-divergence are linked to a word. The grammar of the language is learned as a probabilistic state machine, where each state is a word class and the transition probabilities are derived from the text.

Production and generalization: When describing a topic within a scene, the agent is presented with the topic to describe. The agent uses the state machine to decide which word class should be next, and then a naive Bayes classifier is used with the Gaussian categories to decide on the best word within that class. Using the state machine, DESCRIBER presents complete sentences to describe a shape by its properties.

Characteristics: The novelty of DESCRIBER is the ability to generate descriptions from grounded language through state machines and the use of KL-divergence for handling referential uncertainty. DESCRIBER is able to use KL-divergence with a greedy search to handle referential uncertainty across the dimensions and form categories containing multiple dimensions.

The DESCRIBER architecture has several limitations: it grounds only in perception, so would be unable to ground spatial or temporal terms, and it learns from multiple trials offline. Additionally DESCRIBER is a software agent, and therefore not embodied. The scenes presented to DESCRIBER are simplistic and noiseless, so it is unclear if DESCRIBER's framework could be applied to embodied agents.

2.4.4 Grounding language in actions with the iCub

The *iCub* robot is a humanoid platform (and associated simulator) that is intended for studying infant mental development (Metta et al., 2008; Tikhanoff et al., 2008). Studies using the iCub simulator are state-of-the-art in grounding language in actions (Marocco et al., 2010; Tikhanoff et al., 2011; Stramandinoli et al., 2011; Stramandinoli et al., 2012). These studies use neural networks to associate words with actions and are particularly notable for demonstrating how words can be grounded in the sequences of states that form an action.

Private grounding: Recurrent neural networks provide the private grounding for action representations. Neural networks were used to teach a simulated iCub about object manipulation (Marocco et al., 2010) or grasping and reaching (Tikhanoff et al., 2011). Recurrent neural-networks were used to learn object manipulation and grasping, and a feed-forward network was sufficient to learn reaching. For object manipulation, neurons in the input/output layer consist of encoder values, a roundness value and linguistic values. The recurrent network was able to learn the sequences of encoder values and the roundness values that correspond with the different simulated objects that the network is trained on. For grasping and reaching, the networks learned just the motor control commands required to perform the tasks.

Social grounding: The social grounding was performed differently in each study. In the object manipulation study the linguistic input of the recurrent neural network was used to link a word to an action (Marocco et al., 2010). The linguistic input of the network was set to a value for either *roll*, *slide* or *resist* at the beginning of each of the training patterns and the recurrent neural network learned to associate the linguistic symbols with the sequences of movements. In the grasping and reaching study, a separate feed-forward neural network was used to link words and vision to actions to be performed (Tikhanoff et al., 2011). Verbal input was therefore linked to visual representations of actions.

Characteristics: These studies are of interest for their different ways of grounding words for actions in neural networks: one directly in a recurrent sequence of motor commands (proprioception-like) and the other in sequences of visual states. Grounding actions share some of the same difficulties as grounding temporal terms, as they require grounding in cognition, and they can be difficult to direct attention to. Referential uncertainty was partially handled by speech preprocessing, and partially by the neural network, which learns the associations between words, actions and objects. The main limitations of the iCub studies are that they cannot perform one-shot learning and online learning, and that the production capabilities are limited. For object manipulation, the iCub was able to produce a term to describe an action, but it was not clear how the terms would generalize. For the other actions, the iCub only responded to voice commands, and did not describe actions.

2.4.5 Generalized Grounding Graphs

The Generalized Grounding Graphs (G3) framework is the state-of-the-art in the recognition of natural language directions (Tellex et al., 2011). The framework is able to ground objects, places, paths and events and use the grounded representations for inferring and following routes and performing directed tasks.

Private grounding: The G3 framework features a language parser that is able to translate language statements into symbolic classes called Spatial Description Clauses (SDCs). SDCs refer directly to events, objects, places and paths, each containing different numbers of arguments. The SDCs are linked to their respective meanings, which are modeled in different ways. Events (which make up actions) are modeled as sequences, objects are learned from Flickr images (Kollar et al., 2010), and places and paths are grounded in the semantic map (which is generated using Gmapping (Grisetti et al., 2007)).

Social grounding: The framework models the distribution $p(\Phi|\Lambda, \Gamma, m)$where Φ acts like a lexicon table, defining the link $\phi_i \in \Phi$ between a language construct $\lambda_i \in \Lambda$ and grounding $\gamma_i \in \Gamma$. Data is presented to the framework with descriptions so that the framework can create models for the different object types.

Characteristics: Through the combination of parsing language into SDCs and linking the SDCs to models, the G3 framework is one of the most advanced systems that uses symbol grounding. The limitations of the framework are the lack of one-shot and online learning. Because the data used for grounding comes from many different sources – not all of which are necessarily obtainable by a robot – it is impossible for the robot to continue to learn online. The agents in the project do not produce utterances, so production and generalization is not required and therefore not implemented.

2.4.6 Lingodroids

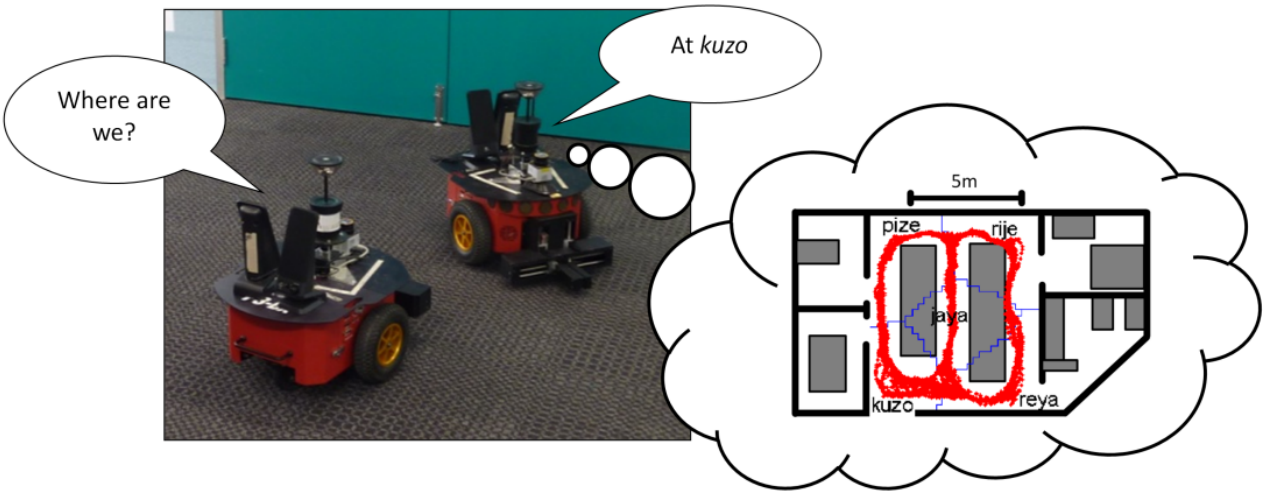


Figure 2.6: Two Lingodroids (embodied as Pioneer 3-DX robots) have a *where-are-we* conversation (reproduced from http://www.lingodroids.org).

Lingodroids is a project at UQ that has been running for eight years investigating robots evolving language for space and time (Schulz et al., 2011a). Lingodroids provides a framework for symbol grounding using shared experiences to share attention and conversations for social grounding. Grounded terms can then be tested using language games as metrics to evaluate the ability of robots to perform practical tasks using their learned terms.

Lingodroids (embodied by Pioneer-3DX robots, see Figure 2.6) were able to create terms for locations (called toponyms) using *where-are-we* conversations (Schulz et al., 2011a). The resulting lexicons were compared using a coherence measure to show that the terms referred to similar regions for each robot. The terms were also tested and used for a practical task by playing *goto* games, where one robot told the other robot to go to a location. The *goto* game is considered a success if the two robots arrive at the same location within a small amount of time.

The Lingodroids could use the conversations *how-far* and *what-direction* to use previously learned toponyms to bootstrap distances and directions (Schulz et al., 2012). The learned distances and directions

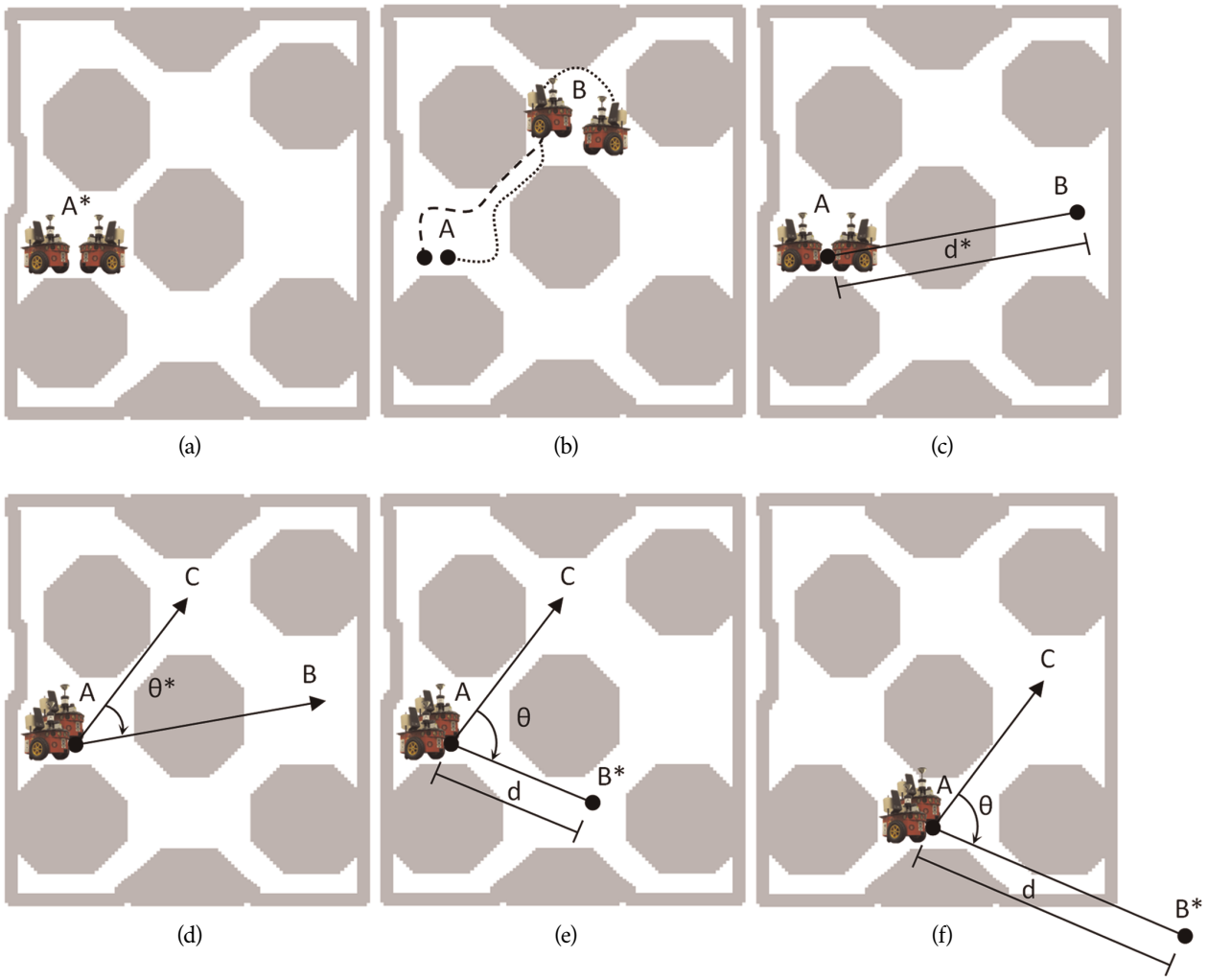


Figure 2.7: Conversations in Lingodroids (reproduced from Schulz et al. (2012)). Values marked with * are learned in conversation. Conversations are a) *where-are-we?* for naming toponyms, b) *goto* for deciding on a toponym to visit (no learning is performed), c) *how-far?* for naming distances, d) *what-direction?* for naming directions, e) *where-is-there?* for generative grounding of toponyms (i.e. naming a toponym that is specified as a distance and direction relative to another toponym), and f) *where-is-there?* for generative grounding of toponyms that have never been visited before.

could then be used to bootstrap further toponyms using another conversation, *where-is-there*, in a process called generative grounding (see section 2.3.1). Using *where-is-there* conversations, the Lingodroids could refer to and name places that they had never been to before.

The Lingodroids project also examined time in simulation (Schulz et al., 2011b). Terms were learned by two robots for the duration of a shared journey between two previously named toponyms. The durations could be taken from an actual journey, or the robots could estimate the duration and generate terms for their estimates. A summary of the conversations and games is presented in Figure 2.7.

Private grounding: Private grounding in Lingodroids uses the RatSLAM cognitive map. Spatial categories are grounded using nodes within the maps and distances and directions are grounded using multiple nodes.

Social grounding: Lingodroids uses conversations for social grounding. Conversations are labeled after the question or activity that is asked or performed, such as *where-are-we*, *goto* or *how-far*.

A typical Lingodroids conversation consists of a question and a response, in the following steps:

1. The two robots repeat the word "hello" until they can both hear each other. At this point they are considered to be in the same place.

2. The first robot to say "hello" is considered the speaker. The speaker asks the listener a question, such as "where are we?", or "how far?".

3. The speaker provides extra parameters for the question. For example, in the *how-far* conversation, two locations are given as the speaker is inquiring about the distance between these locations.

4. The listener chooses the best word to answer the question, or creates a new word. This word is the response given to the speaker.

5. Both robots update their lexicons with the word and place.

In contrast to the Talking Heads (Steels, 2015), no explicit feedback is given after a conversation, but the dynamics of many conversations leads to coherence between multiple agents.

Production and generalization: The Lingodroids' lexicon table is similar to that of the Talking Heads (Steels, 2015), but instead of linking words to categories, it directly links words to *concept-elements* - a vector of features in perception or cognition. KDEs are used to dynamically form distributions when a word is required in a conversation. The distributions are evaluated to find the best word to describe a given feature. Concept elements can either be grounded in RatSLAM maps, or as numbers (for distances, directions and durations).

Characteristics: The Lingodroids project allows grounding of spatial and temporal terms in cognitive maps, therefore allowing grounding in cognition. The spatial terms created cover toponyms, distances and directions. Temporal terms are more limited than the spatial terms – they were only learned in simulation, and were not tested on any practical tasks. The Lingodroids use lexicon tables similar to those of the Talking Heads, which are designed to allow one-shot and online learning.

The limitations of the Lingodroids concern how they deal with different cognitive architectures and referential uncertainty. The Lingodroids handle referential uncertainty using the questions in language games. The question "where are we?", for example, limits the robots' comprehension of the answer to spatial terms; however, this also limits the dimensions that the Lingodroids can refer to. The Lingodroids agents are identical in all studies, so they do not have to deal with different cognitive architectures.

2.4.7 Summary of frameworks

A set of notable frameworks were presented, which in some cases are the state-of-the-art for language learning. However, there are clearly limitations for each study when it comes to a general framework for language learning on mobile robots. The original criteria for evaluation (as mentioned in the introduction)

included: i) ability to ground in cognition (indirectly grounding in perception), ii) ability to ground across sensory and cognitive differences, and iii) ability to solve referential uncertainty. All of the studies handled at least one aspect, but none solved all (see Table 2.1). Studies that were able to ground in cognition used SLAM representations (Jung and Zelinsky, 2000; Schulz et al., 2011a; Tellex et al., 2011) and state sequences (Marocco et al., 2010; Tikhanoff et al., 2011; Tellex et al., 2011) as referents. Words for locations, routes and landmarks were grounded in maps (Jung and Zelinsky, 2000; Schulz et al., 2011a; Tellex et al., 2011). One study handled grounding across sensory differences, but the robots were cognitively identical, and the language models were simplistic (Jung and Zelinsky, 2000). Several studies solved aspects of referential uncertainty (Roy, 2002a; Fontanari et al., 2009; Tellex et al., 2011; Steels, 2015); however, all were limited in embodiment and ability to perform one-shot and online learning.

Table 2.1: Summary of properties

Characteristic	Talking Heads	Cleaning Robots	DESCRIBER	iCub	G3	Lingodroids
Grounding in cognition	-	yes	-	yes	yes	yes
Sensory and cognitive differences	-	yes	-	-	-	-
Referential uncertainty	yes	-	yes	yes	yes	-
One-shot learning	yes	yes	-	-	-	yes
Production of terms	yes	yes	yes	yes	-	yes
Generalization of terms	yes	-	yes	yes	-	yes
Online learning	yes	yes	-	-	-	yes
Human-robot interactions	yes	-	yes	yes	yes	-
Robot-robot interactions	yes	yes	-	-	-	yes
Type of agent	embodied stationery	embodied mobile	software	simulated humanoid	embodied mobile	embodied mobile, simulated mobile
Number of agents	> 2	2	1	1	1	2

2.5 Summary

This chapter has reviewed symbol grounding and many associated problems, solutions and notable frameworks. The symbol grounding problem was theoretically explored in detail during the 1980s and 90s

and the philosophies of Peirce (1974), Searle (1980) and Harnad (1990) have helped to shape the way the problem is approached in modern studies. Studies using embodied robots have demonstrated how embodied agents can autonomously learn grounded symbols, and presented requirements for symbol grounding, including conversations, shared attention, categorization and lexicon tables (Steels, 2015).

Steels claimed that the symbol grounding problem was solved (Steels, 2008); however, his solution did not address i) grounding transfer (Cangelosi et al., 2000), ii) dealing with referential uncertainty (Roy, 2002b; Smith et al., 2006), iii) grounding in cognition (Schulz et al., 2011a, 2011b) and iv) grounding across different sensors and cognition (Jung and Zelinsky, 2000).

For this thesis, literature was reviewed with respect to the aim of creating a framework capable of learning terms for different types of time, learning terms across different sensors and cognition and dealing with referential uncertainty. While several studies have looked at grounding in cognition, only the Lingodroids project has looked at robots developing grounded spatial and temporal concepts in cognitive maps using embodied agents with production, generalization, one-shot learning and online learning (Schulz et al., 2011b). In this case, terms for time were limited to the durations of shared journeys. Many other types of time exist in natural language, and for robots to be able to understand time, other representations and concepts are required. In the next chapter a new framework is presented that borrows the core concepts from the Lingodroids framework, but extends these concepts to handle the limitations of the notable frameworks presented.

Chapter 3

Lingodroids 2: A new framework for autonomous lexicon learning

---

In this methodology chapter, a new framework, Lingodroids 2 (L2) is introduced. The chapter describes the platforms, processes and representations that are required for learning lexicons using the L2 framework, which is used across all the studies in this thesis. As each of the studies in this thesis are self-contained publications, this chapter presents additional methodology details, which are not required to understand the following chapters (Chapters 4-7) or general discussion (Chapter 8), but are included for completeness and may be useful for replication. This methodology omits study-specific processes and representations, as they are described in detail within each study chapter.

The new language learning framework, L2, was implemented to handle the different types of groundings required by mobile robots. The Lingodroids framework was selected as a starting point for the new framework as it:

- is state-of-the-art in robot spatial language learning;
- provides an integrated SLAM system for robot navigation; and
- already allows grounding terms in cognition.

The Lingodroids framework was originally written for the Pioneer 3-DX platform. L2 was rewritten from scratch for a new robot platform for the following reasons:

- to remove dependencies on the Pioneer's hardware;
- to remove dependencies on the original RatSLAM program, which in turn also had dependencies on the Pioneer's hardware; and
- to allow L2 to be more easily extensible: particularly so that conversations, features and transports (transmission capabilities) could be easily added.

The L2 framework retains the core Lingodroids features from grounding in spatial cognition, but provides the processes and representations required to allow grounding in different cognitive architectures and learning through XSL. Between the old and new capabilities, the L2 framework requires many components:

- mobile robot platforms, navigation behaviors and environments;
- distributed lexicon tables;
- conversations and transmission mediums;
- shared attention;
- quality measures; and
- spatial and temporal cognition.

The following sections in this chapter look in turn at each of these components, and then in the final section (Section 3.7) the complete software architecture of L2 is outlined.

3.1 Robot platform and environment

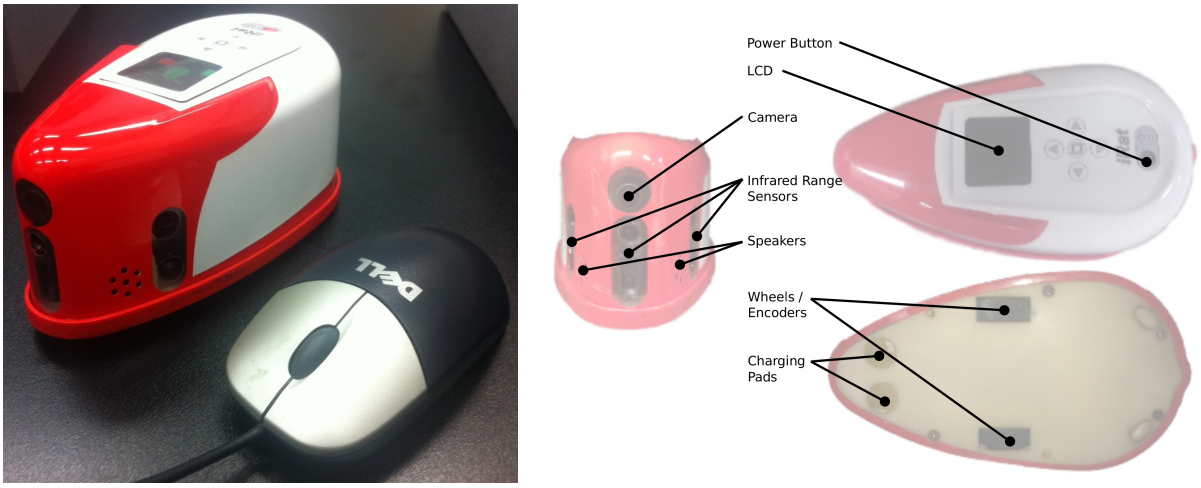
The iRat was selected as the L2 robot platform for the following reasons:

- it is a mobile robot,
- it is small - an environment for the iRat can be contained in a lab,
- it has enough processing power to run the Lingodroids framework, and
- it has appropriate sensors for exploration and navigation.

The iRat has the functional capabilities of a PC on wheels (see Table 3.1). It has a rodent-inspired robot shape that is about the same size and weight as a large rat (see Figure 3.1) (Ball et al., 2010). The robot has IR sensors for avoiding obstacles and a wide-angled forward-facing camera for performing visual SLAM (see Table 3.2 for data ranges for each sensor, and Figure 3.2b for a camera picture).

The iRat was developed at UQ, intended for rat-robot interaction studies (Wiles et al., 2012), but it has also been used for research into spiking-neural networks (Wiles et al., 2010), and as a long-term, autonomous telerobot (demonstrating the stability of the platform) (Heath et al., 2011).

The iRat has some capability for extensibility – additional sensors can be added and exchanged, but usually only when the iRat's case is removed. In Gibson et al. (2014), the iRat's forward-facing camera was replaced with the DVS sensors of Lichtsteiner et al. (2008) in order to collect event-based, high-temporal-resolution datasets (see Figure 3.2c and d). For Studies III and IV in this thesis, the forward-facing camera was replaced by a Hokuyo range-finding laser scanner (see Figure 3.2e and f).



(a) (b)

Figure 3.1: The iRat (intelligent rat animat technology) robot. The iRat is a small mobile robot designed for rat-robot interaction studies. a) the iRat next to a computer mouse (reproduced from Ball et al. (2010)), and b) the iRat features labeled.

Table 3.1: iRat specifications

Category	Specification
Size	170mm long
Weight	0.6kg
Processor	1GHz i586 Vortex 86DX
RAM	256MB
Operating System	Ubuntu 10.04
Hard disk	8GB μSD card
Connectivity	miniPCI 802.11g WLAN card
Battery	7.4V at 2.6Ah

Table 3.2: Physical sensors and actuators

Sensors / Actuators	Data Range
Forward facing camera	416x240 RGB images
Three Sharp IR sensors	0.1-0.4m
Wheel encoders	0-0.5m/s and 0-2rad/s
Motors	0-0.5m/s and 0-2rad/s

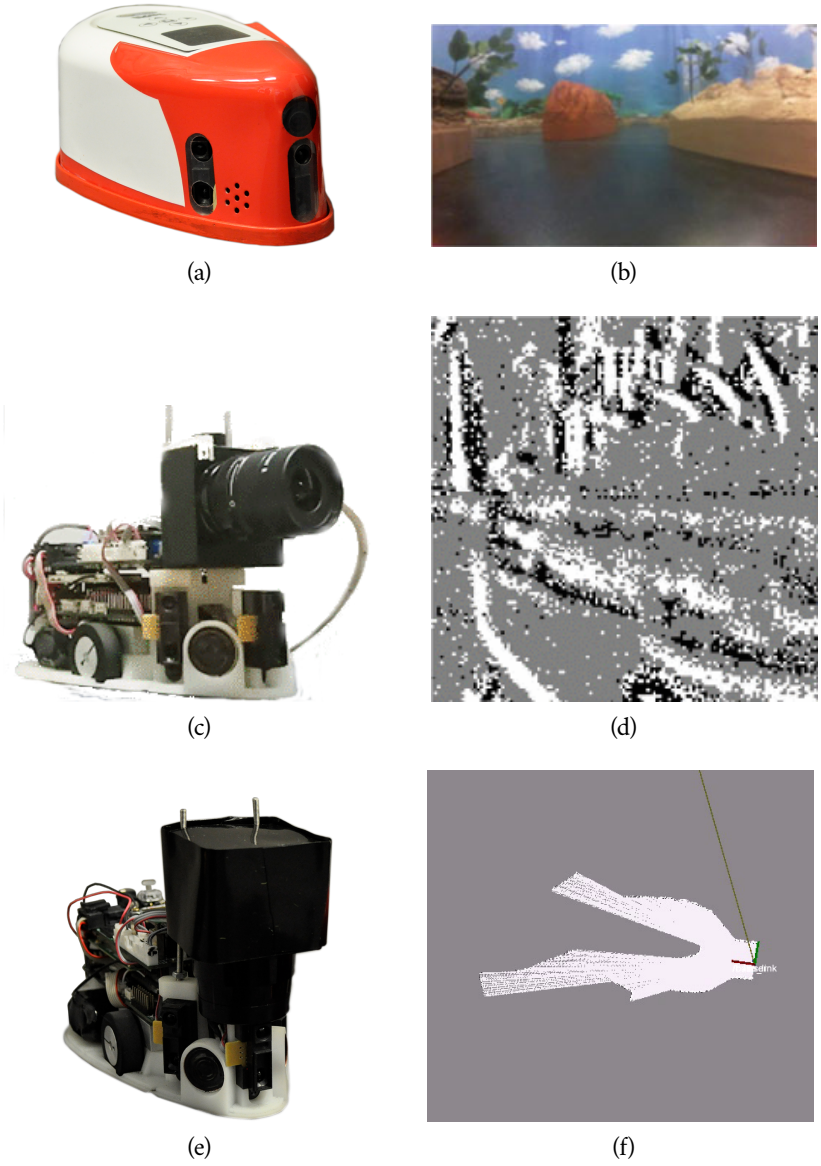


Figure 3.2: Extensions to the iRat and a representation of the associated input. a) the original iRat, b) an image taken with the iRat's forward facing camera, c) the iRat mounted with the DVS of Lichtsteiner et al. (2008) (reproduced from Gibson et al. (2014)), d) a representation of an image taken with the DVS (reproduced from Gibson et al. (2014)) – white pixels indicate positive brightness changes, while black pixels indicate negative changes within a small time window, e) the iRat mounted with a Hokuyo laser scanner, and f) a laser scan representation.

3.1.1 Robot exploration and navigation

Simple robot exploration and navigation behaviors were implemented for all the L2 studies. Both exploration and navigation are implemented similarly to previous iRat studies (Ball et al., 2010; Ball et al., 2010; Heath et al., 2011). Wall following behaviors are used for both exploration and navigation. The exploration and navigation are controlled by a state machine that allows obstacle avoidance, wall following or center following.

Table 3.3: iRat controller

State	Trigger	Assignment
Obstacle avoidance	$D_L < K_O$ or $D_R < K_O$ or $D_C < K_O$	$v = 0$ $\omega = \begin{cases} \omega_{max} & \iff W = \text{LEFT} \\ -\omega_{max} & \iff W = \text{RIGHT} \end{cases}$
Center following	$D_L < K_C$ and $D_R < K_C$	$v = v_{max}$ $\omega = (D_L - D_R) \times K_P$
Left wall following	$W = \text{LEFT}$	$v = v_{max}$ $\omega = (D_L - K_D) \times K_P$
Right wall following	$W = \text{RIGHT}$	$v = v_{max}$ $\omega = -(D_R - K_D) \times K_P$
Goal behind	$G = \text{BEHIND}$	$v = 0$ $\omega = \begin{cases} \omega_{max} & \iff W = \text{LEFT} \\ -\omega_{max} & \iff W = \text{RIGHT} \end{cases}$

D_L	–	distance from the left ranger	v	–	velocity control
D_R	–	distance from the right ranger			($v_{max} = 0.1$)
D_C	–	distance from the center ranger	ω	–	angular velocity control
K_O	–	obstacle avoidance threshold			($\omega_{max} = 1$)
K_C	–	center following threshold	K_D	–	desired distance to wall
W	–	follow LEFT or RIGHT			(usually $K_D = 0.2m$)
G	–	goal is (in FRONT or BEHIND)	K_P	–	proportional gain

During exploration, the iRat randomly changes between following the left and right walls, allowing the iRat to eventually explore all of an environment. The use of center following allows the iRat to retrace the same path very precisely when the path is narrow enough. Retracing allows easier matching of images and increases SLAM performance.

Navigation uses a RatSLAM map, so can only happen once the map has been constructed. The navigation is identical to that described in Heath et al. (2011) (see Table 3.3):

1. A goal is set in the RatSLAM experience map.

2. Dijkstra's algorithm is used to find the shortest path between the goal and the robot's last known position in the map (Dijkstra, 1959).

3. An experience is chosen that is near the robots position, but towards the goal along the route.

4. The relative angle (θ) between the robot and the experience is calculated and normalized to between $-180°$ and $180°$.

5. W and G are set depending on value of θ:

$$W = \begin{cases} \text{LEFT} & \iff \theta > 0° \\ \text{RIGHT} & \iff \theta < 0° \end{cases},$$

and

$$G = \begin{cases} \text{FRONT} & \iff \text{abs}(\theta) < 90° \\ \text{BEHIND} & \iff \text{abs}(\theta) > 90° \end{cases}.$$

6. The set of steps are repeated. Dijkstra's algorithm is not recalculated, the robot's new location is used with the previous path.

In a previous study, it was shown that the navigation algorithm achieved 66% success (Heath et al., 2011). This number is low compared with the large numbers of successes in RatSLAM navigation tests (Milford and Wyeth, 2010). The reason for this difference was the different sensors on the robot platforms used – the Pioneer's omni-directional camera provided better localization during navigation than that of the forward facing camera of the iRat, and the Pioneer's extra sonar range sensors provided richer information about obstacles than the three IR sensors of the iRat.

3.1.2 Robot Operating System

The iRat uses Robot Operating System (ROS) to provide its software interfaces (Quigley et al., 2009). A ROS application consists of sets of executable modules, where ROS provides: i) network communication between modules, ii) abstract interfaces for accessing modules, iii) a large library of modules which provide drivers for common hardware and algorithms for common tasks, and iv) *rosbag* – a utility, associated libraries and format for storing and replaying messages from a running ROS application.

ROS interfaces are provided through topics - an abstraction over network communication. Each topic has a name and an associated binary protocol. ROS modules subscribe to and publish on different topics. The iRat controllers can be run offboard or onboard, using ROS to communicate between the controller and the iRat drivers. If the controllers are run offboard, then the communication will be over WiFi. The iRat subscribes to the topic `/irat/serial/cmdvel`, to receive command velocities and publishes on topics `/irat/serial/odometry` (for velocities), `/irat/serial/rangers` (for IR ranges) and `/irat/camera/images` (for camera images) (see Figure 3.3). The protocols are specified through message descriptions (see Figure 3.4, b-d).

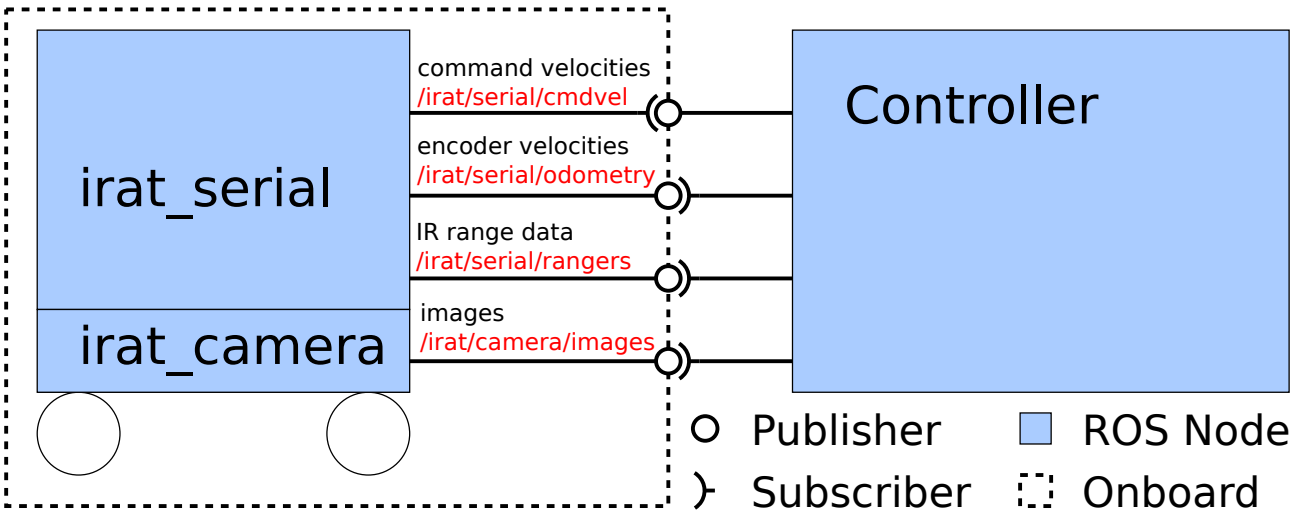


Figure 3.3: iRat interfaces provided through ROS – the connection between the iRat and a controller. The iRat's drivers are provided by two modules: irat_serial and irat_camera. irat_serial provides access to all devices connected to the iRat's real-time micro-controller, which include the motors and IR rangers. irat_camera provides access to the iRat's front-facing camera. Any controller that can provide and receive the iRat's inputs and outputs can be used with the iRat. The topic names are written in red. Controllers connected to the iRat can optionally receive encoder velocities (defined in IRatVelocity.msg), images (defined in CompressedImage.msg) and IR ranger data (defined in IRatRanger.msg) (see Figure 3.4 for message definitions). The command velocity is also set using IRatVelocity.msg

```
# IRatVelocity.msg
# ----------
Header header
float64 magnitude
float64 angle
bool stall
```

(a)

```
# IRatRangers.msg
# ----------
Header header
IRatRanger[] rangers

# IRatRanger.msg
# --------
float64 field_of_view
float64 min_range
float64 max_range
float64 range
```

(b)

```
# CompressedImage.msg
# ------------
Header header
string format
uint8[] data
```

(c)

Figure 3.4: ROS message definitions for iRat interfaces – a) IRatVelocity: the message used for command velocity and odometry, b) IRatRangers: the message used for sending the iRat's ranger data, and c) CompressedImage: the message used for sending camera images from the iRat (CompressedImage is included as part of ROS).

3.1.3 Environments

Two environments are used for the studies in this thesis - temporal cognition studies (Chapters 4 and 5) use the UQ maze, while the different cognitive architecture studies (Chapters 6 and 7) use the Australia maze. Both environments have an associated overhead camera that captures images of the entire area. Both environments were designed for iRat studies and feature narrow "corridors" that allow the iRat to center and retrace its paths. The two environments and simulator are briefly described below.

UQ maze environment: The UQ maze is constructed from a "U" and a "Q" (the initials of the University of Queensland) cut out from a 55 mm high foam sheet (see Figure 3.5). The UQ maze was adapted from a previous study (Heath et al., 2011). The maze was designed so that the interlocking "U" and "Q" create three different loops allowing for odometry information from the iRat to be corrected by loop-closure when using visual SLAM. The UQ maze was also designed to fit within a laboratory, covering an area of 2.07×1.87m. Several objects were added to provide "interesting" visual features to be used for SLAM correction.

Figure 3.5: The UQ maze environment. The UQ maze consists of an interlocking "U" and "Q" to create loops that are a suitable for correction using SLAM. Visually distinct objects are added to the UQ maze to assist SLAM.

Australia maze environment: The Australia maze was designed by UQ's school of journalism as an environment for the iRats. The maze takes up 2.5×1.8m in the lab and is full of visually rich features, some of which are prominent Australian landmarks. The Australia maze contains five interconnecting inner loops, which allow SLAM systems to work effectively (see Figure 3.6).

Simulated Australia maze environment: For Study IV, a simulated version of the Australia maze was used. The simulated Australia maze is comprised of an outline and meshes within the Stage simulator (Vaughan, 2008). Stage provides the robot sensors including: camera, range-finders, laser scanners and wheel odometry. The camera sensor provides medium-fidelity renders of the simulated environment

Figure 3.6: The Australia maze environment (reproduced from Ball et al. (2013)). The Australia maze was designed by UQ's School of Journalism and Communication for iRat SLAM studies. The Australia maze measures 2.5×1.8m

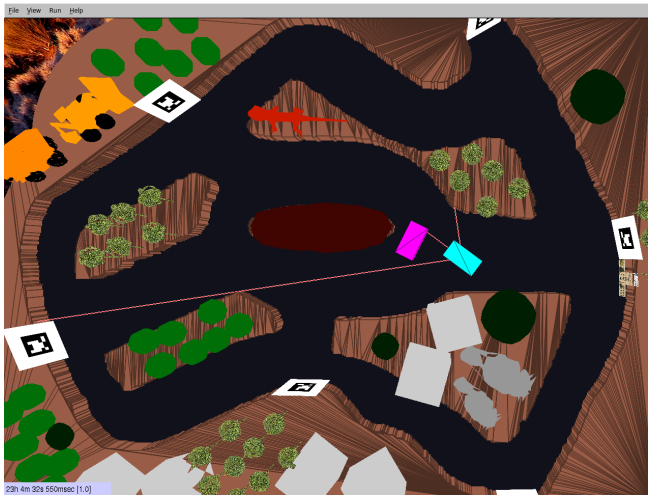


Figure 3.7: The simulated Australia maze is designed to look as much like the actual Australia maze as possible. It situated within the Stage robot simulator (Vaughan, 2008). Mesh-loading additions were added to allow 3D visual cues to be placed within the environment.

which are suitable for SLAM. Using a simulated environment allows a much larger number of conversations to be held, without issues caused by the robot's battery life or connectivity.

3.2 Distributed lexicon table

The L2 framework uses a similar distributed lexicon table to the previous Lingodroid studies. The distributed lexicon table is implemented as a dynamically re-sizable matrix (see Figure 3.8), which links words to concept-elements through many-to-many relationships (Schulz et al., 2011a). Concept-elements are sensory or cognitive samples that are grouped together through their links to the same word to form concepts. A key feature of the distributed lexicon table is the separation between the evidence for a concept and the just-in-time concept use. All the equations given in this section have been used in previous Lingodroids studies (see one of Schulz et al. (2011a), Schulz et al. (2011); or Schulz et al. (2012) for more

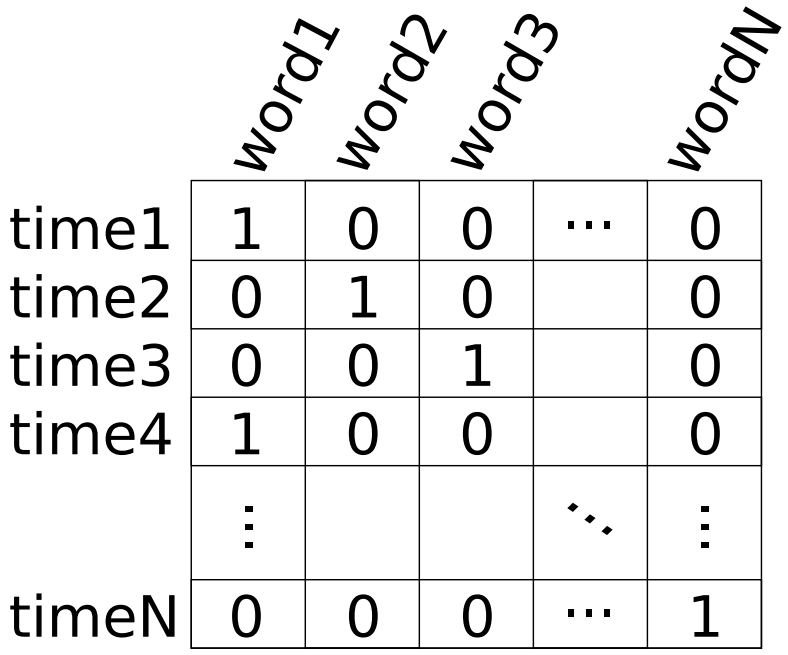


Figure 3.8: The internal structure of the Lingodroid lexicon table: a matrix with non-zero values representing associations between words, and concept-elements. The times are durations that are specified as double-precision, floating-point numbers.

details) and are reproduced again in each of the studies in this thesis. They are also reproduced here for completeness.

An association between element i and word j is updated as follows:

$$a_{ij}{}^{*} = a_{ij} + 1, \tag{3.1}$$

where a_{ij} is the previous association of concept element i and word j and $a_{ij}{}^{*}$is the updated association.

For word production, agents find the word with the highest confidence for the feature that they are trying to name. The confidence value h_{ij} for concept element i and word j is as follows:

$$h_{ij} = \frac{\sum_{m=1}^{Y} \frac{a_{mj}(D - \text{DIST-BETWEEN}(i,\, m))}{D}}{\sum_{n=1}^{N} a_{nj}}, \tag{3.2}$$

where D is the neighborhood size - a constant that defines the maximum distance that a word may be generalized, Y is the number of concept elements in the neighborhood of element i, N is the total number of concept elements, a_{ij} is the association between element i and word j and DIST-BETWEEN $(i,\, m)$ is the distance between concept element i and m, that is calculated using Euclidean distance. Equation 3.2 defines the competition dynamics between different words by expanding concept-elements into a distribution using a KDE. In these studies the constants D are fixed to different values for different dimensions and robots – different values are outlined in the respective studies.

3.2.1 Word invention probability

Words are invented with a probability based on the confidence of the best word given by:

$$p = k \exp\left(\frac{-h_{ij}}{(1-h_{ij})\,T}\right), \tag{3.3}$$

where $k = 1$, h_{ij} is the confidence of the best concept element for a given word and T is the temperature - an adjustable learning rate that was decreased linearly from 0.1 to 0.0 during studies. This probability defines an exponential drop from $p = 1$, when $h_{ij} = 0$ to $p = 0$ when $h_{ij} = 1$. The learning rate T adjusts how fast the drop is (see Figure 3.9). A word will always be invented if there is no word that can be generalized to describe a feature.

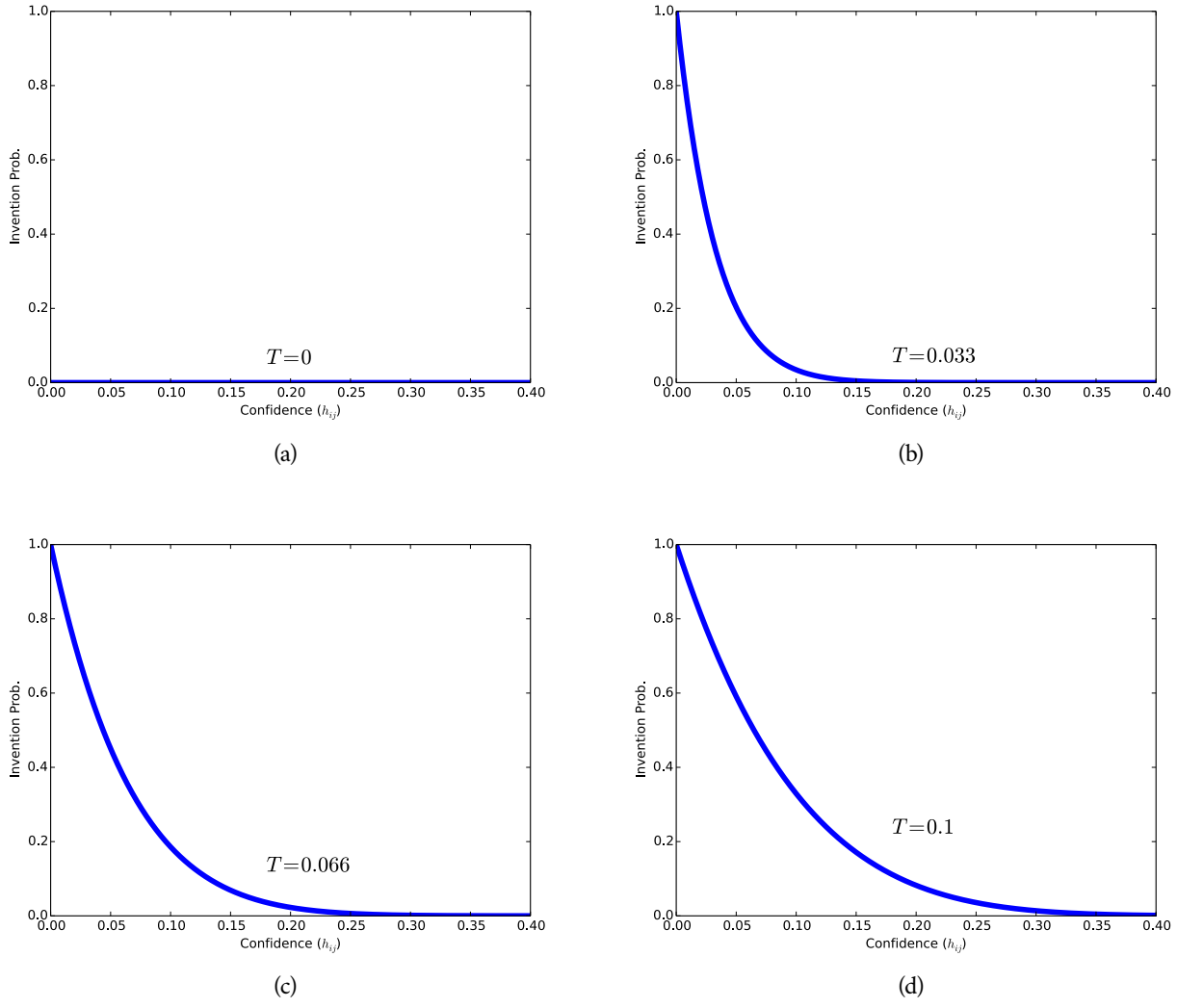


Figure 3.9: The word invention probability for four different temperatures. a) for a temperature of zero, the probability of invention is always zero. b-d) for temperatures greater than zero, the probability starts high for a feature with low confidence and rapidly decreases as confidence increases.

3.2.2 Word production, invention and comprehension

To produce a word for a feature i, the word with maximum confidence (j_{best}) is selected across all of the words j:

$$j_{\text{best}} = argmax_j \left(h_{ij} \right). \tag{3.4}$$

The word invention probability p is then calculated from $h_{ij_{\text{best}}}$ (see Equations 3.2 and 3.3). A number, X, is chosen at random from the range $[0, 1)$ and compared with p. When $X > p$, the word $j_{\text{best}}$is used to describe i. If $X \leq p$ then a new word is invented to describe i. New words are invented by uniformly, randomly selecting two consonants (c_1 and c_2), and two vowels (v_1 and v_2). The consonants and vowels are concatenated into the string "$c_1 v_1 c_2 v_2$", which becomes a new two-syllable word (e.g. "kuzo").

Word comprehension is performed as a neighborhood search across all the concept elements associated with a word. The concept element with the highest confidence (i_{best}) is selected:

$$i_{\text{best}} = argmax_i \left(\sum_{k=1}^{Y} h_{kj} \right), \tag{3.5}$$

where concept element k is within the Y concept elements that are within the neighborhood (see constant D in Equation 3.2).

3.2.3 Visualizing lexicons

Lexicons can be visualized by sampling the confidence of words uniformly using a preset resolution. For the 1D temporal terms, the following steps are performed:

1. A step size dt is chosen (usually $dt = 0.01$) and a maximum duration that the robots refer to D_{max} ($D_{max} = 120$ in this case),
2. For every word j in the set of words used, and for every duration i, from 0 to D_{max} in steps of dt, calculate confidence h_{ij} (see Equation 3.2).
3. For every word j plot all (i, h_{ij}).

The confidence is added as the second dimension (see Figure 3.10).

For the spatial terms, the robot's map is overlaid onto the lexicon, with each term shown as a region within space. The following steps are performed:

1. A step size dt is chosen ($dt = 0.0005$), a minimum and maximum x and y that refer to the minimum and maximum positions in the robot's map.
2. For every x from x_{min} to x_{max} in steps of dt and for every y from y_{min} to y_{max} in steps of dt, do the following:
   (a) For every word j in the set of words used, calculate confidence h_{ij} (see Equation 3.2), where $i = (x, y)$.
   (b) Take $j_{best} = argmax_j \left(h_{ij} \right)$

(c) Draw color $color(j_{best})$ at (x, y) within the lexicon image.

3. For every map node, draw the node onto the lexicon image.

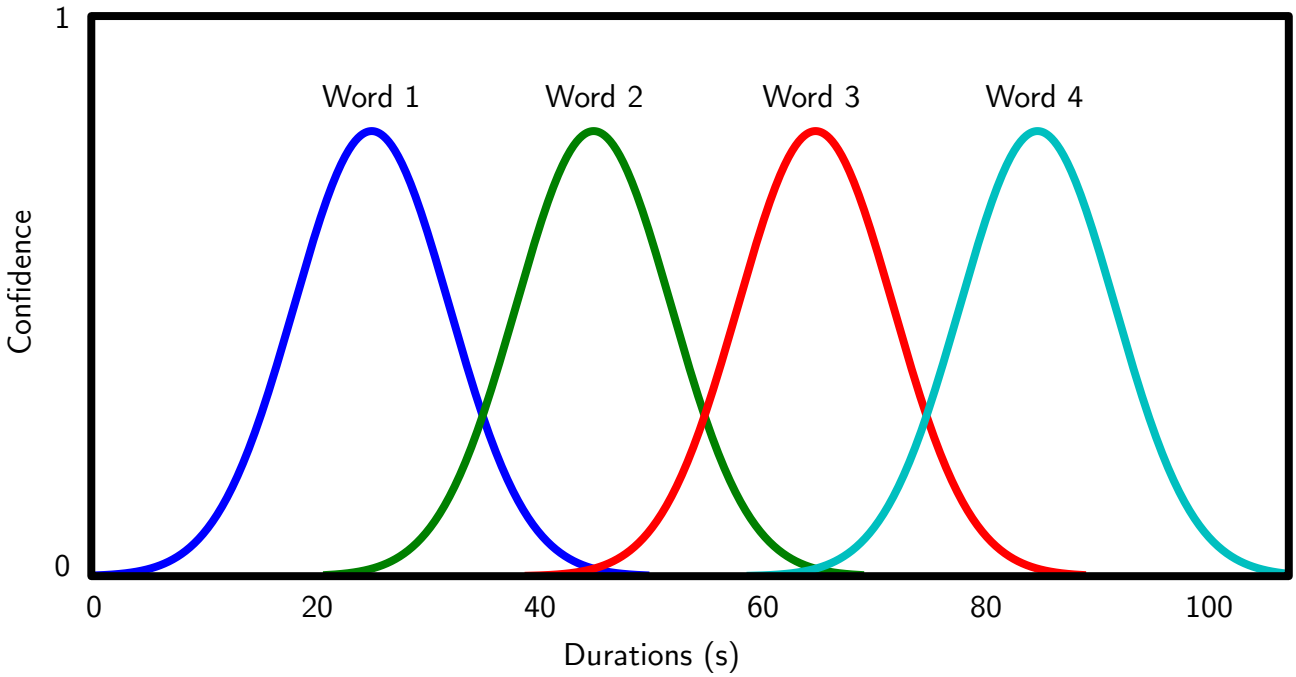


Figure 3.10: Visualization of a temporal lexicon. Each different colored line represents a different word. The X axis is the set of durations that each word refers to, and the Y axis is the confidence of each word about a particular duration.

As there is no dimension to show the confidence, instead the best word is plotted for each region (see Figure 3.11).

A final type of lexicon visualization, is that of cyclic terms. Terms for sunlight levels and angles are plotted in this way. To create the cyclic visualizations:

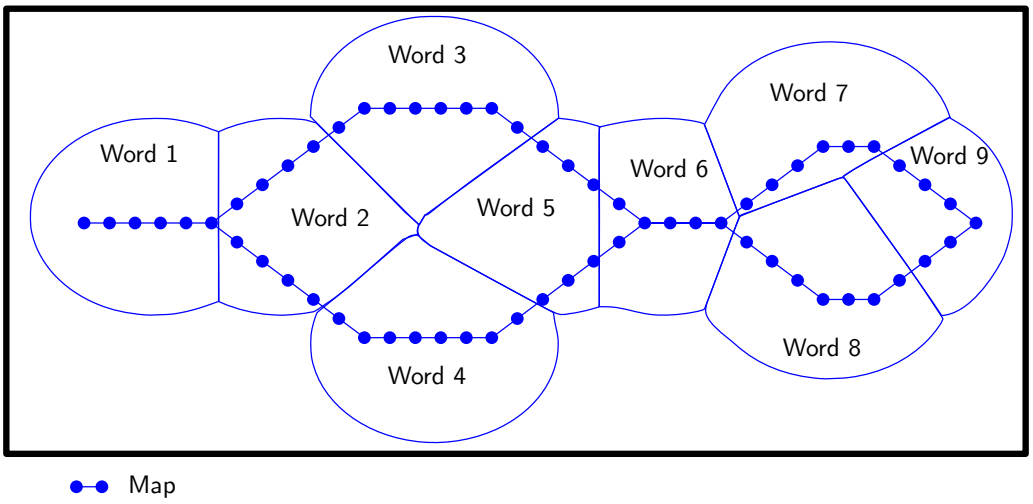
1. A step size is chosen ($dt = 0.002$ was used for these figures), and t_{max}, the largest time in the cycle.

2. For every word j in the set of words used, and for every θ from $0°$ to $360°$ in steps of $\Delta\theta = \frac{dt}{t_{max}} \times 360$, calculate confidence h_{ij} (see Equation 3.2), where time or angle $i = \frac{\theta}{360} \times t_{max}$.

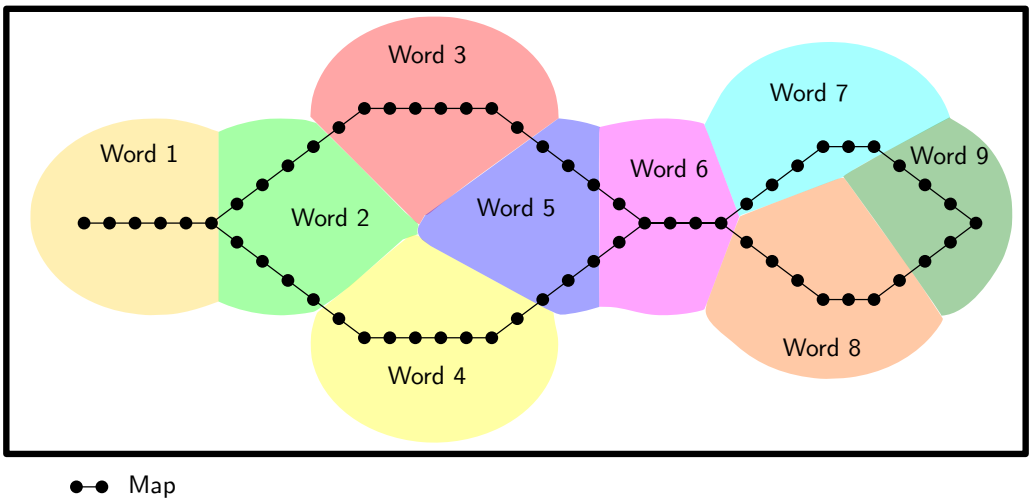
3. Either:

   (a) For the direction lexicons developed by the *what-direction* conversation (see Section 3.3.4): calculate the confidence for the figure using $x = h_{ij} \times cos(\theta)$ and $y = h_{ij} \times sin(\theta)$ and then plot all (x,y) (see Figure 3.12a); or

   (b) For the time of day lexicons developed by the *what-time-of-day-is-it* conversation (see Section 3.3.3): calculate the maximum confidence for $j_{best} = argmax_j(h_{ij})$ and draw a wedge between θ and $\theta + \Delta\theta$ that is color $color(j_{best})$ (see Figure 3.12b).

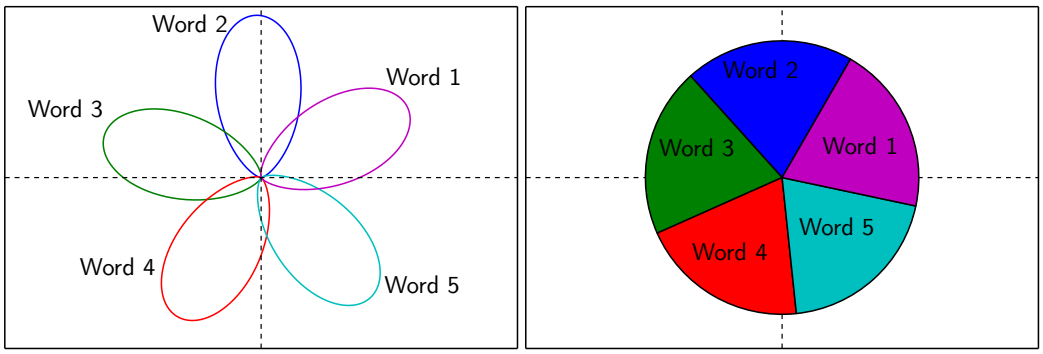


(a)



(b)

Figure 3.11: Visualization of a spatial lexicon. The map is imposed onto the image as a set of linked nodes. Each region represents a spatial word. a) Just the boundaries are shown in blue (as used in Chapter 4). b) Each region is shown as a different color (as used in Chapters 5-7).



(a) (b)

Figure 3.12: Visualization of a lexicon for cyclic terms. a) Terms for angles, each curve represents an angle where the confidence is between 0 (the center of the circle) to 1 the radius of the outer circle. b) Terms for times of day where the best word is shown for each period of the day in Cartesian coordinates.

3.3 Conversations

Conversations (or language games as first introduced by Wittgenstein et al. (1958)) have been demonstrated as a partial solution for the symbol grounding problem on mobile robots (Steels and Vogt, 1997). As previously noted (see Section 2.2.1) conversations pose simple questions from a speaker to a listener. The question predicate provides the context for isolating a feature of an experience. Answering the question requires either creating or generalizing a symbol in order to describe the isolated feature.

Conversations require communication between agents. Where the previous Lingodroids used audio to communicate (DTMF tones, see Schulz et al. (2011a)), in the current studies it was more convenient to use the iRat's wireless network to communicate. Text strings were sent directly from one robot's controller to the other's using ROS messages; however, often the controllers were run on the same desktop computer.

Lingodroid papers have included several different conversations (see Figure 2.7). Several of the previous Lingodroids' conversations are used in these studies, and several new conversations are derived from the previous conversations. Specific conversations are used in all the studies in this thesis and are presented in Chapters 4-7 and discussed briefly below.

3.3.1 *where-are-we*

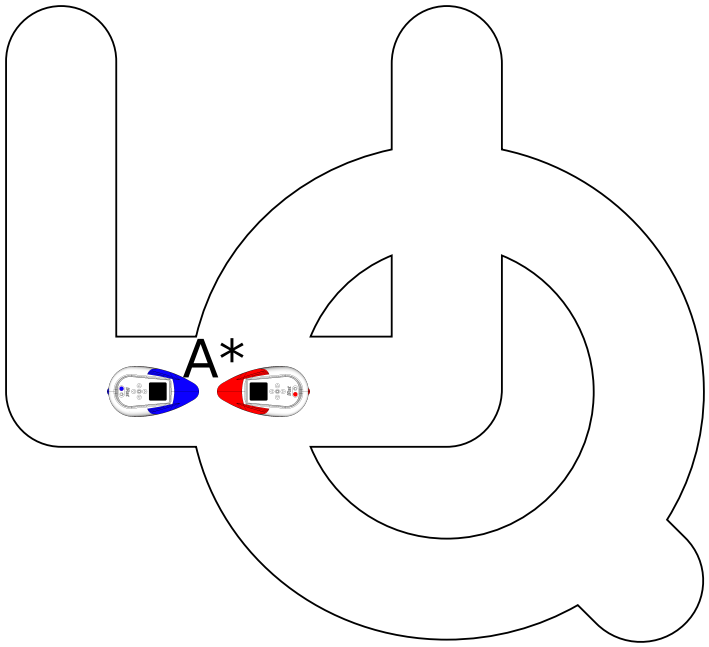


Figure 3.13: The *where-are-we* conversation. One robot asks the other: "where are we" and the other provides a name, A, to describe the place. The $*$ indicates that both robots remember the new word.

The *where-are-we* conversation was introduced in previous Lingodroids studies for developing spatial language (Schulz et al., 2011a). The conversation follows the question and answer steps of Section 2.4.6:

1. Two robots meet in the environment.

2. A speaking robot asks a listening robot "where are we?"

3. The listening robot invents a word to describe the place of the meeting, or generalizes an already existing word. The choice of inventing or generalizing is based on the word invention probability (see Sections 3.2.1 and 3.2.2).

4. Both robots associate the word with their current experience.

The conversation was implemented on the iRats as part of the implementation of the L2 framework (see Figure 3.13).

3.3.2 *when-did-we-last-meet*

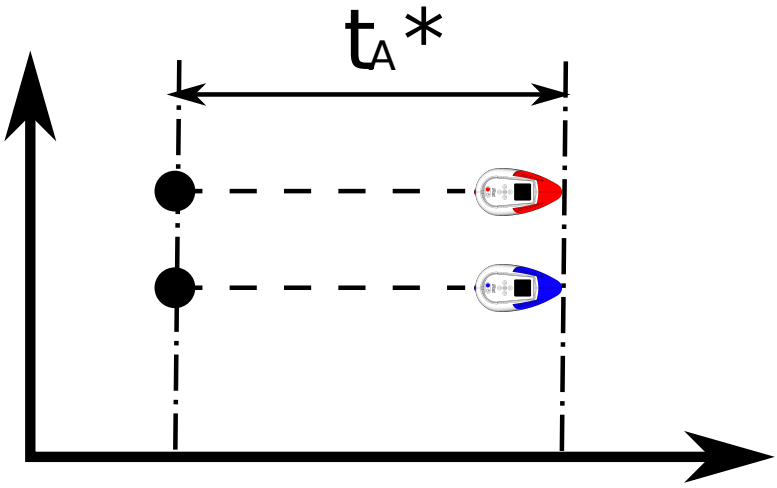


Figure 3.14: The *when-did-we-last-meet* conversation. One robot asks the other: "when did we last meet" and the other provides a name A to describe the duration since the last meeting. The $*$ indicates that both robots remember the new word.

The *when-did-we-last-meet* conversation was introduced in the L2 framework, but was inspired from previous Lingodroids conversations that ground time in cognitive maps (Schulz et al., 2011b). The steps are very similar to the *where-are-we* conversation and are as follows:

1. Two robots meet in the environment.

2. A speaking robot asks a listening robot "when did we last meet?"

3. The listening robot invents a word to describe the duration between the previous meeting and the current meeting, or generalizes an existing word, based on the word invention probability (see Sections 3.2.1 and 3.2.2).

4. Both robots take the difference between the current clock time, and the time recorded at the previous meeting as t_A. Both robots associate the current word with t_A.

One of key differences between the *where-are-we* conversation and the *when-did-we-last-meet* conversation is that the former labels an absolute place, while the latter labels the difference between two absolute times (see Figure 3.14). In this regard, the *when-did-we-last-meet* conversation is more like the previous Lingodroids *how-far* conversation (see Section 3.3.4).

3.3.3 *what-time-of-day-is-it*

The *what-time-of-day-is-it* conversation was introduced in the L2 framework to handle different types of time. This conversation was inspired from the sunlight equation of Meeus (1991) and has similarities to Steels' robots that name event-based scenes (Steels and Baillie, 2003). The conversation follows similar steps to *where-are-we* as follows:

1. Two robots meet in the environment.
2. A speaking robot asks a listening robot "what time of day is it?"
3. The listening robot invents a word to describe the sunlight level, or generalizes an existing word, based on the word invention probability (see Sections 3.2.1 and 3.2.2).
4. Both robots associate the given word with the current sunlight level.

More information is given about this conversation in the relevant study (see Section 5.2.2).

3.3.4 *how-far*

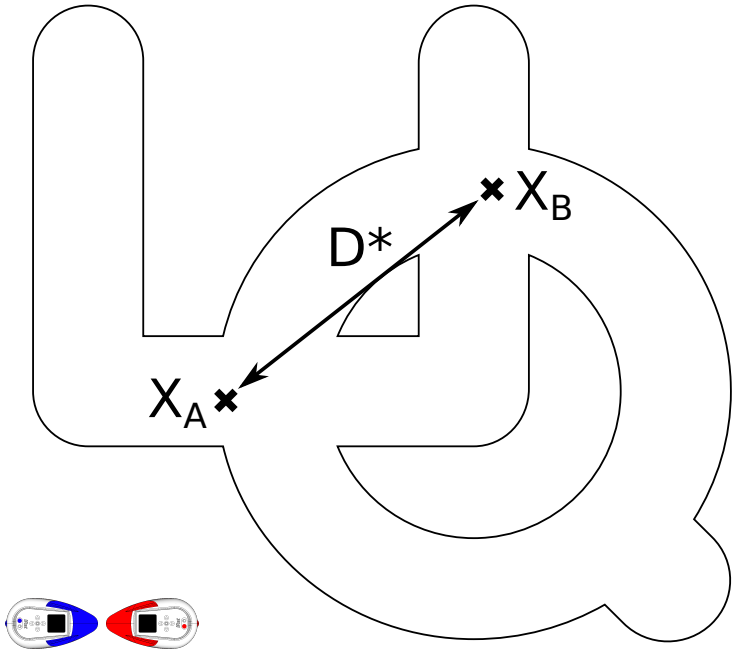


Figure 3.15: The *how-far* conversation. The two robots initiate a conversation but do not require shared proximity, or to be in the environment. One robot asks the other "how far is it between places X_A and X_B". The other robot provides a name D to describe the distance between the two places. The $*$ indicates that both robots remember the new word.

The *how-far* conversation was introduced in previous Lingodroids studies to develop lexicons for distances through grounding transfer (Schulz et al., 2012). The conversation was used within this thesis to test lexicons of robots with different cognitive architectures. The conversation steps are as follows:

1. Two robots initiate a conversation (but not necessarily together in the environment).

2. A speaking robot asks a listening robot "how far?"

3. The speaking robot provides the names of two places that have previously been learned by the two robots.

4. The listening robot invents a word to describe the distance between the two places, or generalizes an existing word, based on the word invention probability (see Sections 3.2.1 and 3.2.2). The distance between the two places is calculated using the robot's cognitive map.

5. Both robots associate the given word with their calculation of the difference between the two places.

The crucial differences between this conversation and the *where-are-we* conversation are that i) the conversation does not require physical proximity in space or time, and shared attention is established using language, and ii) the conversation depends on previously developed spatial lexicons and cognitive maps (see Figure 3.15).

3.3.5 *what-direction*

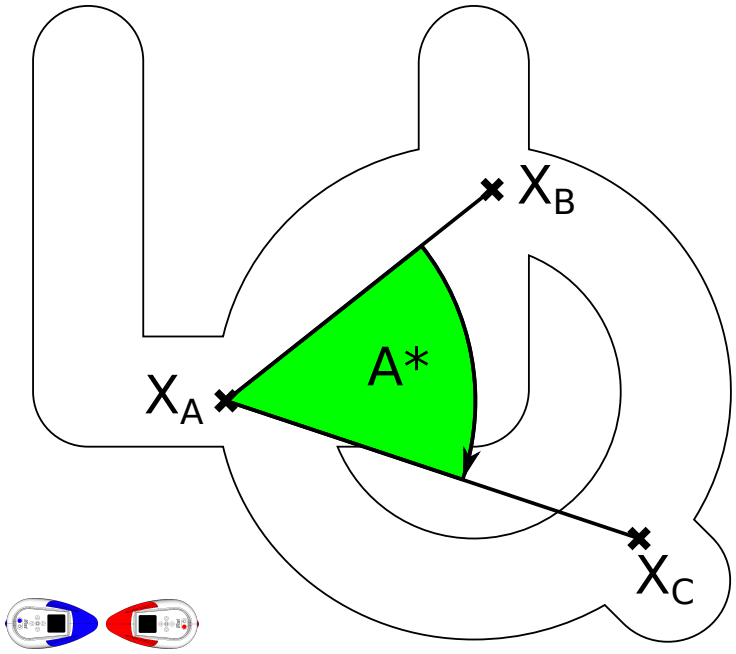


Figure 3.16: The *what-direction* conversation. The two robots initiate a conversation but do not require shared attention, or to be in the environment. One robot asks the other "what relative direction is X_C if I am at X_A facing X_B". The other robot provides a name A to describe the angle between the two places. The $*$ indicates that both robots remember the new word.

The *what-direction* conversation is very similar to *how-far* in that it was also introduced in previous Lingodroids studies to develop lexicons to allow grounding transfer (Schulz et al., 2012). The difference is that *how-far* labels distances that have an underlying linear structure derived from pairs of toponyms, whereas *what-direction* labels angles derived from three toponyms. Like *how-far*, this conversation was used within this thesis to test lexicons of robots with different cognitive architectures. The conversation steps are as follows:

1. Two robots initiate a conversation (but not necessarily together in the environment).
2. A speaking robot asks a listening robot "what direction?"
3. The speaking robot provides the names of three places that have previously been learned by the two robots.
4. The listening robot invents a word to describe a direction based on the three places, or generalizes an existing word, based on the word invention probability (see Sections 3.2.1 and 3.2.2). The direction is calculated using the robot's cognitive map by assuming that the robot is at the first place, A, facing the second place, B. The direction is the angle that the robot would need to turn to instead face the third place, C. The angle, θ, is given by:

$$\theta = atan2(C_y - A_y, C_x - A_x) - atan2\left(B_y - A_y, B_x - A_x\right), \tag{3.6}$$

   where $atan2(y, x)$ is the quadrant corrected $atan(y/x)$.
5. Both robots associate the given word with their calculation of the direction between the three places.

Like the previously described *how-far* conversation, the *what-direction* conversation does not require physical proximity in space or time, and shared attention is established using language (see Figure 3.16).

3.3.6 *where-in-space-time-are-we*

The *where-in-space-time-are-we* conversation was introduced in the L2 framework to handle referential uncertainty. It was inspired by previous robot studies (Steels and Kaplan, 2002; Oates, 2003; Steels, 2015), and other referential uncertainty studies in computer science and psychology (Akhtar and Montague, 1999; Smith et al., 2006; Smith and Yu, 2008; Vogt, 2012).

1. Two robots meet in the same environment.
2. A speaking robot asks a listening robot "where in space-time are we?"
3. The listening robot generalizes 0-2 terms to describe the place and time. The term for place is based on the robot's current location in the cognitive map, and the term for time is the duration since the two robots last met.
   (a) If no generalizations are available to the listening robot (i.e. no terms are provided), the two robots switch roles, and the speaking robot generalizes 0-2 terms.
   (b) If the speaking robot also has no generalizations, the speaking robot invents 2 terms, one for place and one for time, and provides those.
4. The robot that provided the terms associates the terms with their correct features in the cognitive map and in the remembered duration.
5. The robot that received the terms associates all terms with both spatial and temporal features.

More information on the *where-in-space-time-are-we* conversation is provided in the relevant study (see Section 7.3.2).

3.3.7 *meet-at*

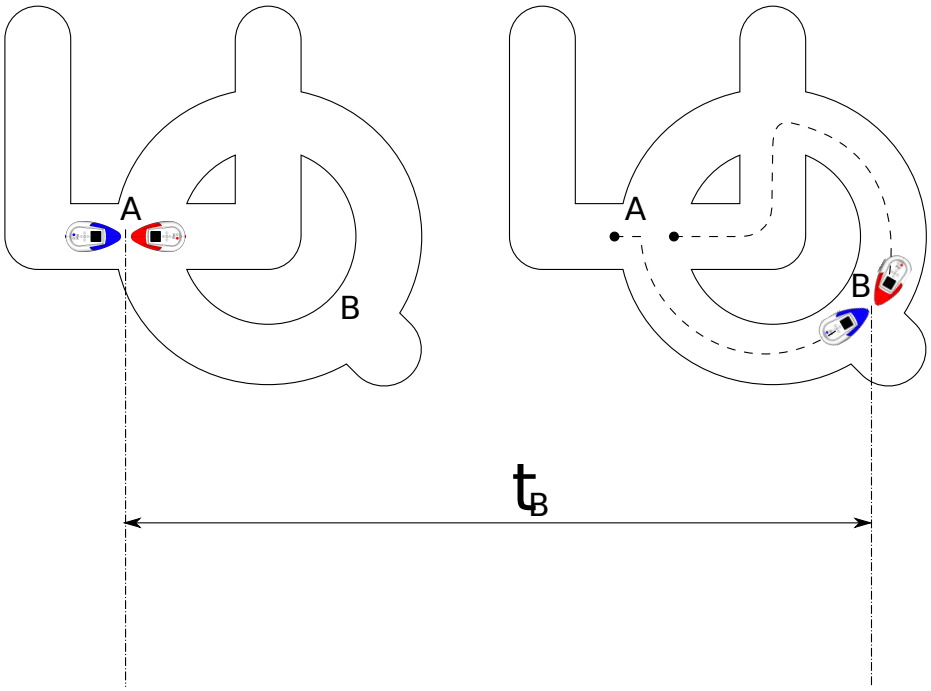


Figure 3.17: The *meet-at* game. Two robots meet at place A in the environment and arrange to meet at place B at time t_B. The robots attempt to navigate to place B within the window of time that falls under the term t_B. No learning is performed within this study.

The *meet-at* game was introduced in the L2 framework to test the Lingodroids' spatial and temporal lexicons. The game was inspired from the *goto* game in previous Lingodroids studies (Schulz et al., 2011a, 2012). The steps of the *meet-at* game are as follows:

1. The two robots meet in the environment.
2. A speaking robot selects a place word B and a temporal word t_B from previously learned lexicons and tells the other robot to "meet at" place B at time t_B. There are two variants of temporal terms used with the *meet-at* game: i) terms grounded in durations, and ii) terms grounded in times of day.
3. The two robots continue exploring until they need to move to the meeting. There are two choices for when to start moving:
   (a) The two robots plan how long it will take them to get to the goal (t_T) using their cognitive maps and wait until time $t_B - t_T$ before starting to move to the goal. In practice the iRat's estimation of time taken, based on previous trip times was not the same as active navigation, and so this method was too inaccurate to work effectively.
   (b) The two robots keep track of the best word for the current time given by:

$$Z_{t_C} = max_j \left(h_{ij} \right), \tag{3.7}$$

across all the words, j, where $i = t_C$, the current time (see Equation 3.4 in Section 3.2.2). The robots then start moving once $Z_{t_C} \equiv t_B$. In practice, this method was more robust than calculating the time taken.

4. When the two robots have arrived at their set place and time, they are assessed by an overhead camera and external clock, which report how far apart the robots are in space and time.

No learning is performed within the *meet-at* game, it is used only for testing previously created spatial and temporal lexicons (see Figure 3.17). The *meet-at* game can only be undertaken after a learning phase in a study, in which spatial and temporal lexicons have been developed using other conversations such as *where-are-we* and *when-did-we-last-meet*.

3.3.8 Conversation implementation

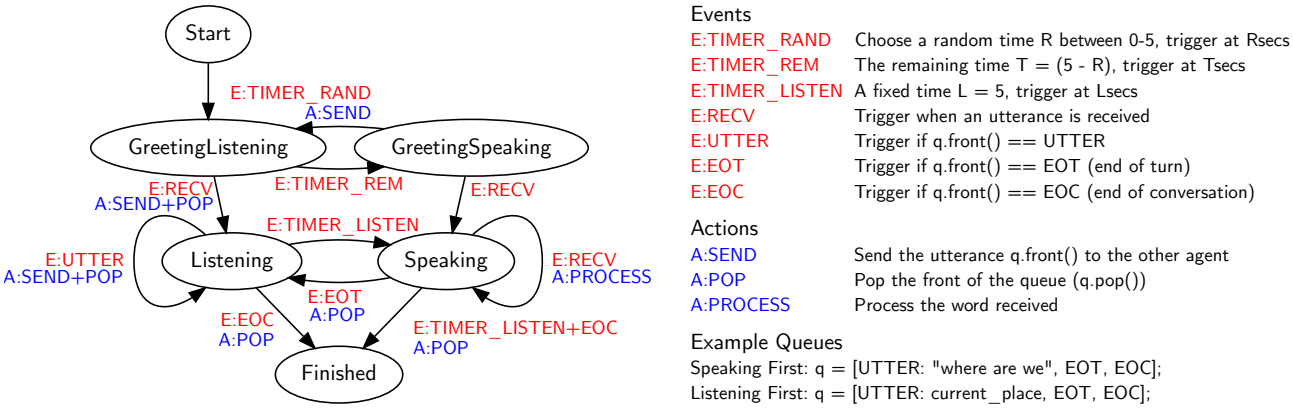


Figure 3.18: The state machine for a learning conversation. This diagram shows the conversation state diagram (left), events that transition between them (in red) and actions that trigger on changes (in blue). The TIMER_* events are all generated from a timer that is reset on receiving the previous event.

Conversations have previously been implemented within Lingodroids as state machines (see Schulz (2008) pg. 33) and they are again in the L2 framework. The conversation state machine and an example conversation are shown in Figure 3.18. The software architecture allows for more complicated conversations to be added easily, as each agent will keep listening until the interlocutor stops speaking. The *Conversation_Controller* class implements the logic within the state machine of Figure 3.18. An implementation of the *Conversation* class then defines the state, lexicons and logic for specific conversations (see Section 3.7).

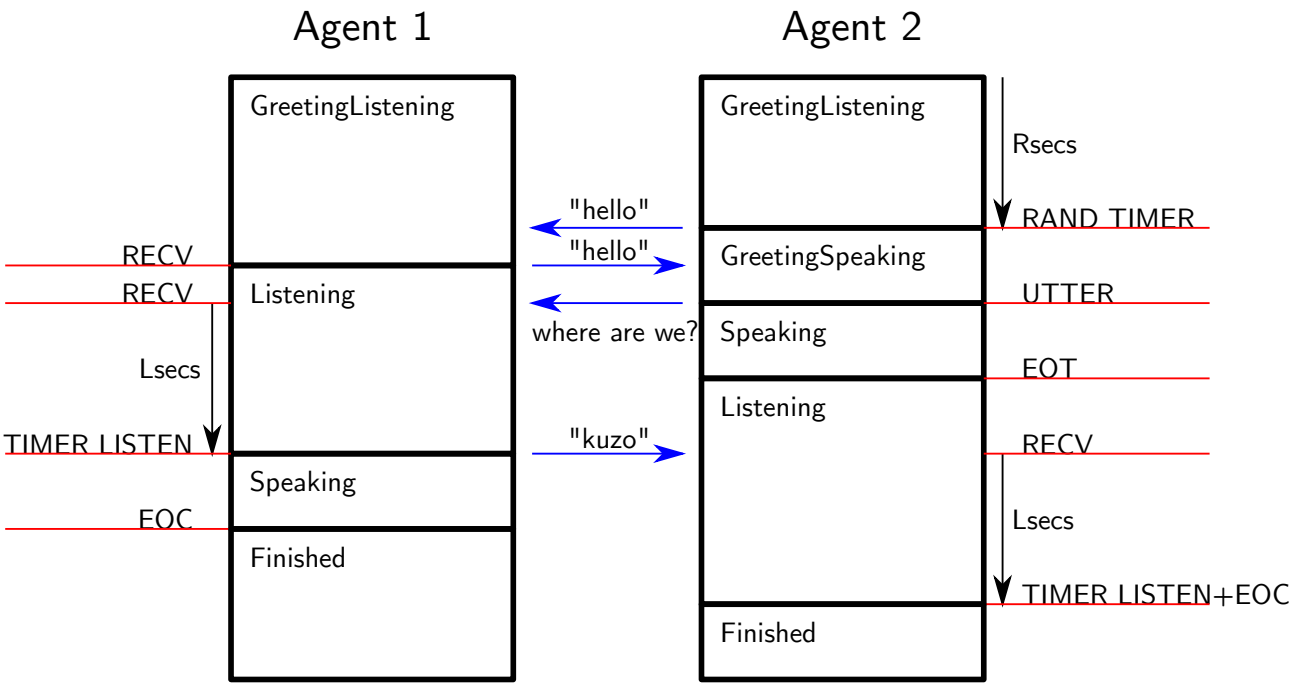


Figure 3.19: A typical successful *where-are-we* conversation between two agents. The agents both start in the GreetingListening state. They each choose a random amount of time, wait for that amount of time and then utter "hello". The first to utter "hello" becomes the first speaker, while the second becomes the first listener.

3.4 Shared attention

As noted in section 2.2.2, shared attention is a requirement for conversations, and to fulfill this requirement *shared experiences* have been used in previous Lingodroids studies (Schulz et al., 2011a). In a shared experience, two agents have internal representations that correspond to the same referent at the same time.

In earlier Lingodroid studies, with bigger environments, shared experience was established by the hearing-distance of two robots - that is the distance over which one robot could hear an utterance from the other (Schulz et al., 2011a; Schulz et al., 2011). This method is unsuitable for the much smaller iRats, as they can hear the other robot anywhere within their environment, which does not allow them to establish the proximity of the other robot.

In these studies, an overhead camera was used to track the iRats and establish shared attention when co-located. The overhead camera provided robots with an entering and exiting signal for shared attention. Within the simulator, shared attention is established by providing a simulated entering and exiting signal depending on the robots' proximities to one and other.

3.5 Quality measures

Quality measures are very important for robot language learning studies, as it can be difficult to tell the difference between coherent and incoherent languages. In Schulz et al. (2011a) coherence of two spatial lexicons is established by aligning the robots maps and then imposing a grid onto the Lexicons. Coherence is then given as:

$$\frac{1}{N_X N_Y} \sum_{x=0}^{N_X} \sum_{y=0}^{N_Y} \begin{cases} 1 & \iff W_1[x,y] \equiv W_2[x,y] \\ 0 & otherwise \end{cases},$$

where N_X and N_Y are the grid x and y dimensions respectively, and $W_1[x,y]$ is agent 1's word at grid coordinate (x,y). RatSLAM maps are semi-metric, which allows for some variation in odometry and image matching at the cost of reduced global accuracy. This means that aligning maps can be a difficult task, although in practice, for previous studies, the maps were similar enough that the alignment was fairly trivial (rotating and translating the maps) (Schulz et al., 2011a). The resolution of the grid can affect the accuracy of the coherence, so small Δx and Δy are used ($\Delta x = \Delta y = 0.25m$ for previous studies in Schulz et al. (2011a), where the world size is large, or $\Delta x = \Delta y = 0.02m$ for the iRat studies, were the world size is always less than $3m \times 3m$).

For temporal, distance and direction lexicons, coherence is calculated in the same way using one dimension. However, no alignment is necessary for these lexicons, as the concept-elements refer directly to absolute values – durations in seconds, distances in meters and angles in degrees.

For robots with different cognitive architectures it is much harder to apply the traditional coherence to the spatial lexicons, as the alignment of maps is not trivial. For these studies, coherence was calculated on either: i) the distances and directions that were bootstrapped from toponyms, or ii) the toponyms transformed back into the robots' shared environment. Both these measures are described in more detail in the relevant studies.

The other major quality measure is the *meet-at* game, described in more detail in Chapter 4. The *meet-at* game sets a spatial and temporal meeting task for the robots, where success is based on the completion of the game (whether both robots were able to find the goal location) as well as the distance between the robots at the goal location and the length of time that one robot had to wait for the other robot. Spatial distances between robots are calculated from the overhead tracking system, while the robots' clocks provide enough accuracy to measure temporal distances. The *meet-at* game is an important quality measure, as it assesses the practical usability of the learned spatial and temporal lexicons.

3.6 OpenRatSLAM

Lingodroid spatial studies (including Schulz et al. (2011a, 2011b, 2012)) have previously used RatSLAM maps for grounding spatial language (Milford and Wyeth, 2010). A suitable SLAM system was required for representing space and grounding spatial language for the L2 framework. RatSLAM was ported to the iRat through a series of iterations:

- a MATLAB version was constructed for iRat, with C optimizations (Ball et al., 2010),
- a C++ version was constructed to allow the iRat to autonomously map and navigate for long periods (Heath et al., 2011),
- the C++ version was refactored into a series of ROS modules, which became OpenRatSLAM (Ball et al., 2013).

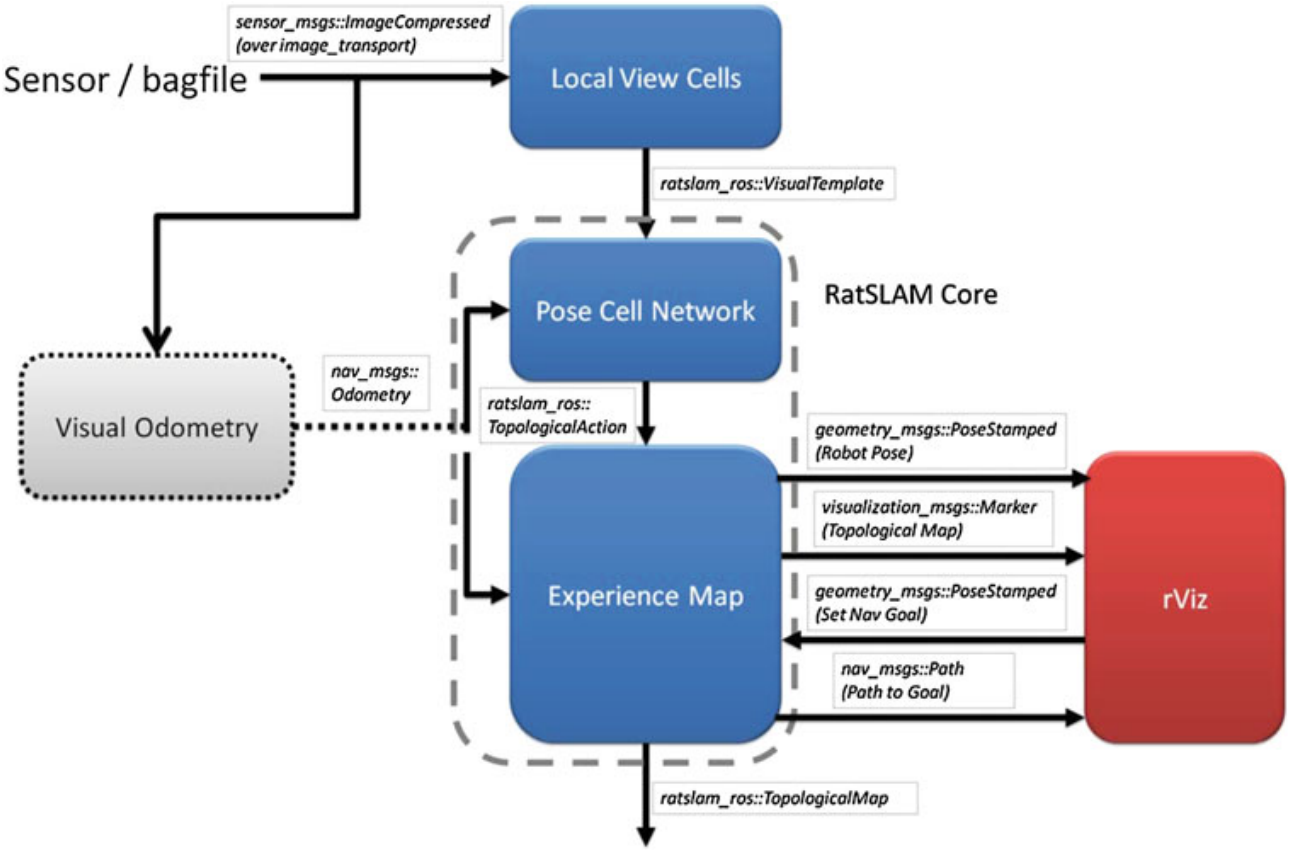


Figure 3.20: The structure of OpenRatSLAM – connections are labeled with their associated ROS messages (image taken from Ball et al. (2013), see also Appendix A). Images and odometry are the input to the three blue RatSLAM modules. The local view cells compare images together and output the nearest matching ID; the pose cell network maintains a continuous attractor network that is manipulated by image matches and odometry; and the experience map creates a human-readable semi-metric map from the instructions given by the pose cell network, and from its own integration of odometry. The visual odometry module is an optional module included as part of RatSLAM that allows images to be used to generate odometry instead of taking odometry from wheel encoders. rViz is a visualizer included as part of ROS.

OpenRatSLAM is an open source version of the RatSLAM algorithms of Milford and Wyeth (2010) arranged into a ROS application. OpenRatSLAM is described briefly here and in more detail in Appendix A of this thesis. OpenRatSLAM is used as three ROS modules within this thesis:

The visual template module: This module receives images from the iRat's camera and matches them against previously seen images. It outputs a template ID – either the ID of a previously matching template or a new ID to assign to the current template.

The pose cell network module: This module receives the template IDs of the visual template module and wheel odometry from the iRat. The posecell network associates the template IDs with cells within a continuous attractor network (CAN). Receiving a previous template ID causes energy to be injected into the associated cells. The wheel odometry is used to shift the energy to follow the robot's motion. The posecell network also stores associations between CAN cells and map nodes (experiences). The outputs from the posecell network are a map action and an experience ID. The actions include creating a new

experience, changing to a new experience and linking them together (a loop closure), or changing to a new experience without linking.

The experience map module: This module receives the map action and the experience ID from the pose cell network, and wheel odometry from the iRat. The experience map module is responsible for maintaining a human-readable semi-metric map by forming and iteratively correcting a graph of linked experiences.

3.7 Software architecture

The Lingodroids software architecture has been written to allow extensibility, particularly for adding additional conversations, features, transmission channels and shared attention mechanisms (see Figure 3.21). The extensibility comes from providing interfaces for conversations, features, word sources, word sinks and shared attention detectors. Conversations can be extended by providing methods to generate requests and process responses. The interface for features is just a single method `abstract_distance_from (other: Feature)`, which allows two features to be compared. The word sources and sinks allow for the implementation of alternate transmission channels, and the shared attention interface allows for the implementation of alternate methods of detecting shared attention.

3.8 Summary

This chapter has described the different invariant components of the L2 framework. The L2 framework is constructed from the core Lingodroids features (distributed lexicon tables, conversations, shared attention and coherence) with extensions to handle advanced groundings of lexicons. A new robot, the iRat, was chosen as the the robot platform for the L2 framework. Simple exploration and navigation algorithms allow the iRat to create maps and set navigation goals using RatSLAM. The implementation presented is extensible to new features, conversations and transmission channels.

The next four chapters present a series of studies, where the L2 framework is applied to particular tasks: learning terms for durations (Chapter 4), learning terms for times of day (Chapter 5), learning terms for toponyms across different cognitive architectures (Chapter 6) and learning terms for toponyms and durations using XSL (Chapter 7). In each of these studies, the L2 framework is tested across two robots on the quality measures of coherence, on practical tasks and on learning time.

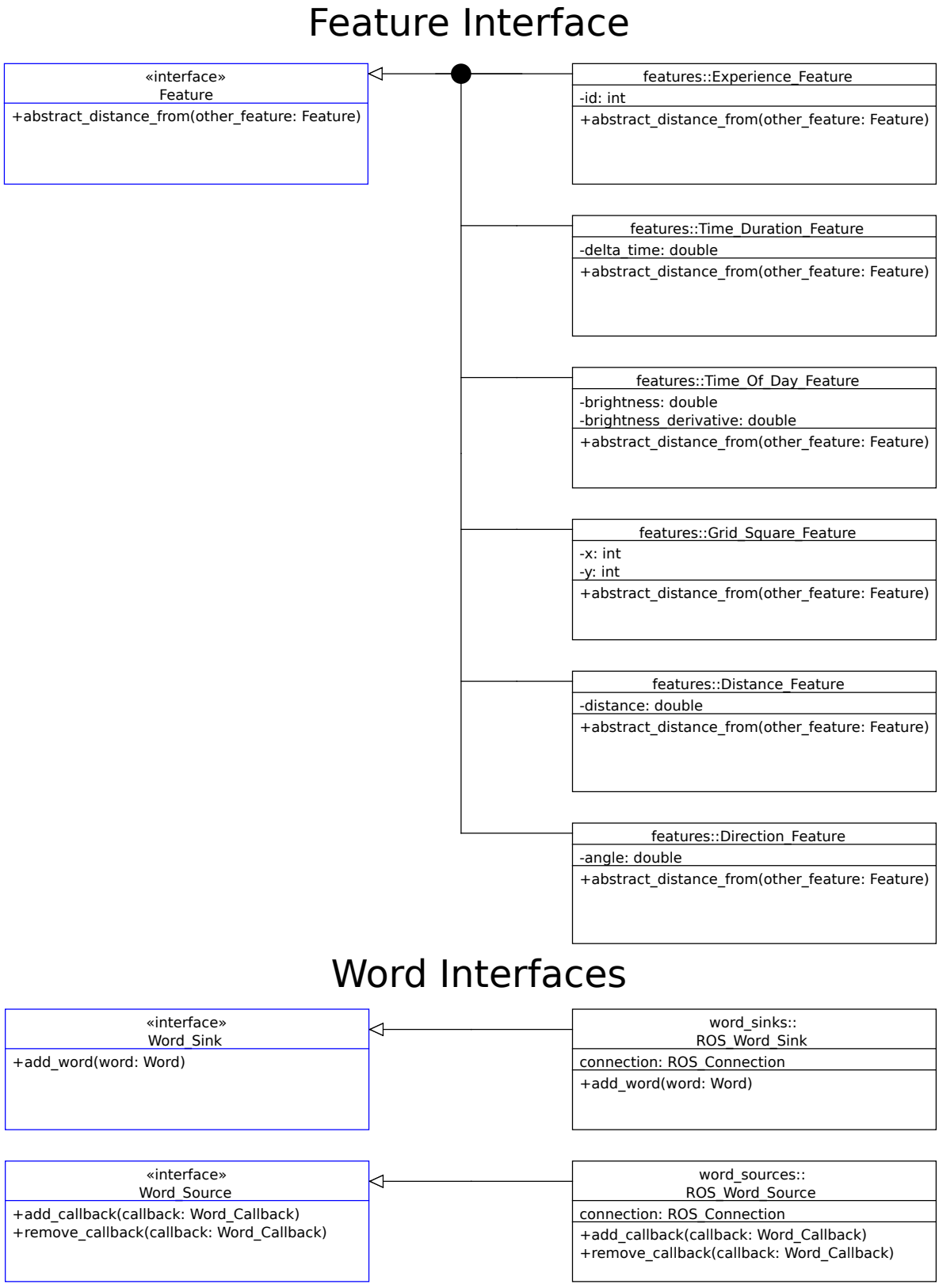
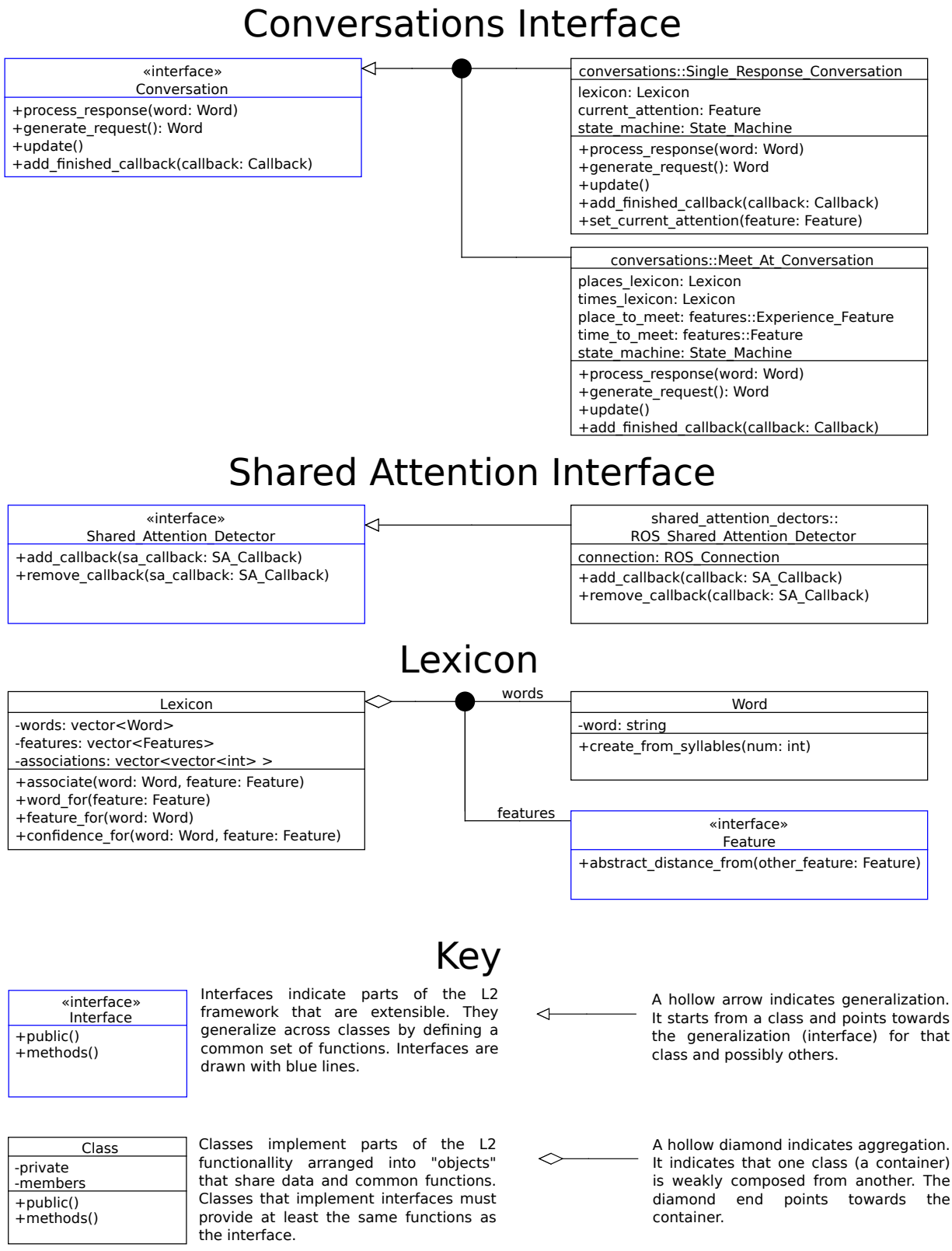


Figure 3.21: The Lingodroids software architecture. The key design abstractions allow extensibility through the interfaces Feature, Conversation, Word_Sink, Word_Source and Shared_Attention_Detector. Implementing the interfaces Feature and Conversation allows new features and conversations to be added.



The implementation of Word_Sink and Word_Source allows the addition of new word transports (e.g. audio). The class Shared_Attention_Detector allows the implementation of new ways of deciding when the robots should be considered to have shared attention.

Chapter 4 – Study I

Lingodroids: Learning terms for time

Temporal cognition and communication are important to mobile robots for scheduling and controlling spatial locomotion. However, as reviewed in Chapter 2, previous research into temporal terms covers only a few types of time and temporal cognition. One of the goals of this thesis is to allow mobile robots to develop spatial and temporal lexicons. Robots can already learn spatial terms grounded in cognitive maps (Jung and Zelinsky, 2000; Schulz et al., 2011a; Tellex et al., 2011), so temporal cognition and communication are required to complement the spatial abilities.

While previous studies have looked at sequencing (Steels and Baillie, 2003; De Beule, 2006) and grounding time in shared journeys (Schulz et al., 2011b), many other natural language constructs of time (see Section 2.3.3 and the included Figure 2.3) have not been studied. In previous temporal learning studies, lexicons were tested by a comparison between agents (see Section 3.5). While a comparison is a reasonable test of similarity, it does not take into account how the words are used (production). No previous studies have used practical tests for temporal lexicons.

Two studies were designed to investigate how mobile robots could learn and test terms for durations. In the first study, the robots developed both temporal and spatial lexicons and tested the coherence of each. In the second study, the robots tested their temporal and spatial lexicons on a new practical task.

A new conversation was designed, *when-did-we-last-meet*, to share and ground duration symbols within temporal lexicons. The terms that the robots developed were grounded in cognition based on clock time and were similar to natural language expressions such as "a short time", "now" or "a long time". The *meet-at* game was introduced as an extension of the *goto* conversation (see Section 3.3 and Section 2.4.6) that requires the robots to coordinate and use both spatial and temporal terms to specify a future meeting. To successfully perform this task, the robots developed both the new duration lexicons and also toponym lexicons using RatSLAM as a cognitive map, like previous Lingodroid studies. The *meet-at* game allowed the robots' spatial and temporal lexicons to be quantified on the success of the task.

The robots in this study were embodied as iRats and the studies took place in the UQ maze (as described in Section 3.1 and shown in Figures 3.1 and 3.5). In the first study, the iRats were allowed to move autonomously around the environment following the exploration algorithm from Section 3.1.1. When the robots had shared attention, they held both a *where-are-we* conversation and a *when-did-we-last-meet* conversation to develop their spatial and temporal lexicons (see Sections 3.3.1 and 3.3.2 for more details on the *where-are-we* and *when-did-we-last-meet* conversations respectively). The lexicons from each robot

were compared and were visualized (see Section 3.2.3). The results of this study demonstrate a coherence of 74.3% across the two robots' spatial lexicons and 80.8% across the two robots' temporal lexicons.

In the second study, the iRats were again allowed to move autonomously around the environment; however, when they met, they instead held a *meet-at* game. 25 *meet-at* games were played, with 14 of the trials ending in success (both robots successfully finding a navigation goal). For the 14 successful trials, the robots' average spatial distance apart was 0.12 meters and the average amount of time one robot waited for the other was 8.6 seconds.

The conclusions of these studies are that the L2 robots are capable of learning duration terms with only minor additions to the core framework required to adapt from spatial terms to temporal terms. The stability of the L2 core is an important feature throughout the thesis. Key insights learned were the additions required to the L2 systems, particularly the grounding of durations in temporal cognition based on events (the last meeting of the robots and the current meeting) and the clock time between events. This temporal cognition enables conversations and groundings that are similar to that of distances and directions. The addition of temporal terms and conversations enables the L2 robots to achieve the thesis goals of communication grounded in spatial and temporal cognition. The conversations introduced in Chapter 4 also enable the learning and use of temporal terms in subsequent studies (see Chapter 7).

The following sections have been reproduced from:

- Heath, S., Schulz, R., Ball, D. and Wiles, J. (2012). Lingodroids: learning terms for time. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1862-1867.

The sections have been taken from the final submitted manuscript and reformatted to fit within this thesis. Note: within this publication, the L2 framework is referred to as just Lingodroids.

Abstract – For humans and robots to communicate using natural language it is necessary for the robots to develop concepts and associated terms that correspond to the human use of words. Time and space are foundational concepts in human language, and to develop a set of words that correspond to human notions of time and space, it is necessary to take into account the way that they are used in natural human conversations, where terms and phrases such as 'soon', 'in a while', or 'near' are often used. We present language learning robots called Lingodroids that can learn and use simple terms for time and space. In previous work, the Lingodroids were able to learn terms for space. In this work we extend their abilities by adding temporal variables which allow them to learn terms for time. The robots build their own maps of the world and interact socially to form a shared lexicon for location and duration terms. The robots successfully use the shared lexicons to communicate places and times to meet again.

4.1 Introduction

Space and time are fundamental aspects of the world. Spatial and temporal concepts are the foundations of an embodied agent's knowledge of the word and are critical for effective inter-agent communication (Kant, 1998). Although time can be measured using clocks to arbitrary levels of precision, natural language terms like 'soon', 'recently', 'later', and 'in a few minutes' are ubiquitous. Such concepts are useful approximations for describing everyday events including the length of time taken to get from one location to another. If robots are to understand natural language they need to learn and understand approximate terms for time that are grounded in experience and can be used in practical tasks.

To date, we are not aware of any studies with robots that can learn and use natural language terms for temporal durations. Other studies have examined the concurrency and sequencing of events (De Beule, 2006) and have constructed temporal ontologies for specific purposes (Hobbs and Pan, 2004; Zhou and Hripcsak, 2007). In this work, we take a different direction and examine how robots can acquire a grounded language for temporal durations.

In all natural languages there is a strong relationship between words for time and words for space. Our work draws on research from fields including psychology and linguistics. In this study we are not directly using spatial concepts to learn temporal ones. Rather, we examine whether methodology that has successfully been used to learn spatial terms can be applied to learning terms for time.

4.2 Related work

In previous work we have demonstrated on real robots that Lingodroids can learn a language to describe space consisting of terms for locations and the spatial relationships between locations (Schulz et al., 2011). The Lingodroids have also formed duration terms in a virtual world, with the underlying representations for the duration concepts based on distance traveled, change experienced, and time taken for a shared journey (Schulz et al., 2011b).

Concepts for space and time have many similarities, with a common theme in different languages being the use of metaphors mapping space to time (Boroditsky, 2000), such as 'time as a path' and 'life as a journey'. The types of metaphor used for temporal concepts include moving ego or moving time (Gentner et al., 2002) and a neutral or specific perspective (Moore, 2006). Representations for temporal concepts have included temporal logics, such as interval algebra (Allen, 1983) and state transition graphs (Clarke and Emerson, 1982). The variety of temporal concepts possible includes durations (of an event or between events), points in time (temponyms, such as a specific date), and the sequencing of events in time (including tense and aspect).

In this chapter, we demonstrate that given conceptual dimensions for space and time, real robots can not only autonomously learn spatial and temporal terms, but also use these terms effectively in a social task: meeting at a particular location in a specified time. We focus on the spatial concepts of locations (toponyms) and the temporal concepts of durations as the basic concepts required for this meeting task.

We present a study involving two Lingodroids, which we have ported for this study to the iRat robot platform. The robots autonomously develop their own cognitive maps of their environment, socially interact to form names for places in the world and temporal durations, and effectively use the spatial and temporal terms to play *meet-at* games with each other. Videos of the Lingodroids are available at www.lingodroids.org.

4.3 Methods

The work presented in this chapter involves a new implementation of the Lingodroids on the iRat – intelligent Rat Animat Technology – robot platform. The Simultaneous Localization and Mapping System, RatSLAM, was ported to the iRat and used for spatial mapping (Milford and Wyeth, 2010). The robot platform, the mapping system, the language system, and the quality measures used are described in the following sections.

4.3.1 Robot platform: iRat

For this study two iRat (Ball et al., 2010) robots played language games (see Figure 4.1). The iRat is a small wheeled robot the same size and mass as a large rat. The iRat has a 1GHz x86 processor (RoBoard), forward facing wide screen camera, three Sharp IR range sensors orientated at -45, 0 and 45 degrees, speakers and a microphone. The robot runs Robot Operating System (Quigley et al., 2009) on Ubuntu and communicates to clients over wireless 802.11g. Two ROS nodes run on the robot, one to publish compressed 416×240

pixel color JPEG images from the robot's forward facing camera at 20Hz, the second to send and receive desired and actual velocity commands.

For this study the iRat uses its IR range sensors for local navigation and obstacle avoidance of the walls and the other iRat. The local navigation module uses a heading direction given by higher level goal planning, or randomly chooses a direction for exploration. In the absence of obstacles the iRat heads directly for a goal if known, otherwise it either follows a wall or the center of a corridor as appropriate.

Figure 4.1: This photo shows two iRat robots meeting to play a language game. The iRats have differential drive (not shown) and a forward facing camera that looks through the top of the 'i'.

4.3.2 Mapping system: RatSLAM

RatSLAM, inspired by the rodent hippocampus, is a SLAM system capable of providing persistent real robot operation (Milford and Wyeth, 2010). RatSLAM creates semi-metric topological spatial representations of an environment using visual and self-motion information. This SLAM system uses appearance based visual and self motion odometry sensor information. The spatial representation consists of experiences that are interconnected by links in a network (called an experience map).

To reach a goal experience, RatSLAM uses Dijkstra's algorithm (Dijkstra, 1959) to determine the route with the minimum duration from the robot's current location. This algorithm uses the duration between the experiences as recorded and stored in the links of the experience map during map formation. The route is optimized for duration rather than distance to take into account factors that may impede the robot such as clutter. The iRat implementation of RatSLAM is a skeleton version of the full system, enabling effective navigation but lacking features such as map maintenance for persistent operation. For full details of RatSLAM refer to Milford and Wyeth (2010).

4.3.3 Language platform: Lingodroids

The spatial concepts for toponyms were formed using methods developed in previous Lingodroid studies (Schulz et al., 2011a): the pair of robots develop a shared language via social interactions called conversations. The first conversation the robots engage in is called *where-are-we*, in which they create names for their current location. After a series of *where-are-we* conversations, the robots have a shared set of names referring to different places in the world – a toponymic language – that can be used to play *go-to* games, in which the robots meet at a remote location.

In a previous study in a virtual world (Schulz et al., 2011b), the Lingodroids formed duration concepts by interacting through *how-long* conversations, in which the length of a journey from one location to a second location was calculated from the distance traveled, the change experienced, or the time taken.

In the current study using real robots, the Lingodroids use a new type of conversation – *how-long-since-we-last-met* – in which the robots calculate the length of time since they last interacted and name this duration. Durations based on the time between robot interactions allow for a larger variety of durations than would be possible from the durations of shared journeys within the robots' environment. A second new type of interaction, an extension of the *go-to* game, is used to test the usefulness of the spatial and temporal lexicons: the *meet-at* game, in which one robot specifies both the location and time of a future meeting. The robots aim to arrive at the specified location at the specified time. Success of the game is measured by how far apart they are and how long one robot has to wait for the second robot to arrive.

The robots interact when shared attention has been established, which is done through the use of an overhead camera to detect when the two robots are within 0.25 meters of each other, corresponding to 80 pixels in the overhead camera's image. This strategy was directly transferred from the virtual world studies, in which shared attention was established when the robots were located within a set distance of each other.

A novel aspect of the Lingodroids is the data structure used to associate the name of a word with its features. Associations between words and concept elements (experiences for toponyms and lengths of time for durations) are stored in two *distributed lexicon tables* (Schulz et al., 2011a), one each for toponyms and durations. A distributed lexicon table allows many-to-many associations between words and concept elements, rather than between a word and a single feature corresponding to its meaning. Concept elements for durations are created when new durations are experienced in the *how-long-since-we-met* conversations. Associations record the number of times a word and concept element are used together in a conversation. Such distributed associations enable word use to be established in a single trial but also to evolve over time with additional trials.

In each conversation, the speaker chooses the word to name the location or duration by finding the confidence value for each word and choosing the word with the highest confidence value. The confidence value, h_{ij}, for a word, j, and a concept element, i, is calculated as follows:

$$h_{ij} = \frac{\sum_{k=1}^{X} a_{kj} \left(D - d_{ki}\right) / D}{\sum_{m=1}^{N} a_{mj}}, \tag{4.1}$$

where X is the number of concept elements within a neighborhood of size D of the current concept element, i; a_{ij} is the number of times the concept element, i, and the word, j, have been used together; d_{ki} is the distance between concept element k and i; and N is the total number of concept elements that have been created by the robot. In this study a 0.4m neighborhood was used for the toponyms and a 15 second neighborhood was used for the durations. Words are invented with probability, p, using the confidence value and a word invention temperature, as follows:

$$p = k \exp\left(\frac{-h_{ij}}{(1-h_{ij})\,T}\right), \tag{4.2}$$

where $k = 1$, h_{ij} is the confidence value of the element-word combination, and T is the temperature. T was set to 0.1 in this study. For full details of the algorithms used see Schulz et al. (2011a).

In a *meet-at* game, the speaker chooses a random experience and a random time up to the maximum duration experienced. A toponym is found for the experience and a duration word is found for the time. Both the speaker and the hearer then determine the best location for the toponym and the best time for the duration word. They each continue to explore until they decide that it is time to go to the goal location, based on how far away from the goal they were. They then switched to goal-directed navigation, and once the goal has been found stop and finish the game. A *meet-at* game fails if one or both of the robots are not able to find the goal before a time-out occurs.

4.3.4 Quality measures

The measures used to determine the quality of the lexicons were the coherence of the lexicons and the success of the *meet-at* games.

Coherence is calculated over a set of locations for toponyms and a set of times for duration words. In this study, the set of locations were at 0.02 meter intervals, measured at the intersections of a grid covering all locations in and around the maze. The set of times were every 0.5 seconds. Coherence provides an indicator of whether the lexicons are similar, defined as the percentage of the conceptual dimension for which the same word was used by both robots.

For the *meet-at* games, success is based on the completion of the game (whether both robots were able to find the goal location) as well as the distance between the robots at the goal location and the length of time that one robot had to wait for the second robot. Distances between robots were calculated from the overhead tracking system.

4.4 Experimental setup

The experiment was conducted using two iRats in a 1.9m by 2.1m maze that included several loops with an interlinking U and Q, standing for The University of Queensland (see Figure 4.2). The toponymic and duration lexicons were formed and used in this environment in two stages:

Exploration and Formation of Spatial and Temporal Concepts: Both robots were placed in the maze, with each robot independently exploring and building individual maps of the world. The robots

Figure 4.2: The maze used in the study as shown from the overhead camera used to establish shared attention between the robots. The blue iRat is in the left side of the U and the red iRat is in the right side of the Q. Visual features have been placed around the edges of the maze to enable the mapping system to create coherent maps of the maze.

interacted whenever they met and established shared attention, playing a total of 30 *where-are-we* and 23 *how-long-since-we-last-met* conversations.

Use of Spatial and Temporal Concepts: After the formation of the spatial and temporal lexicons, the concepts were tested through a series of 25 *meet-at* games.

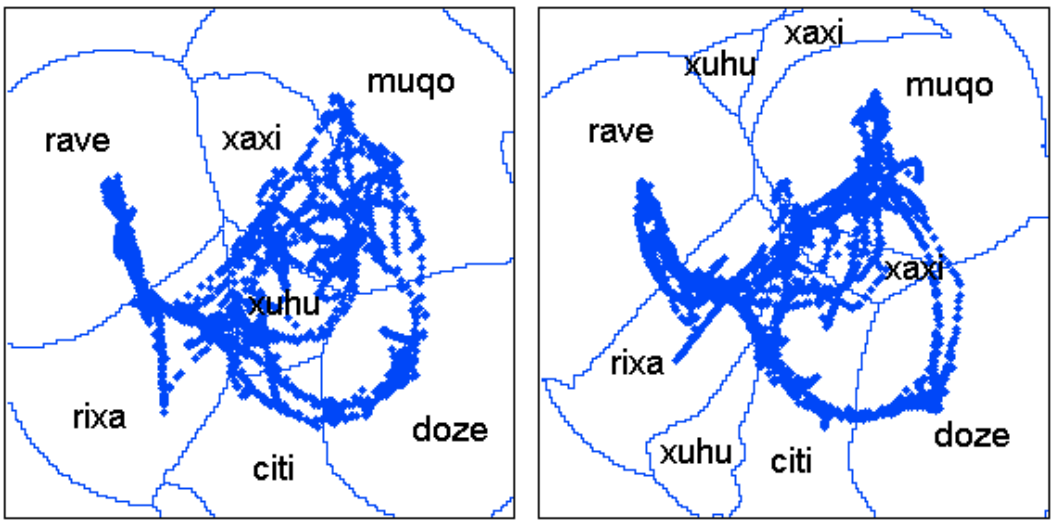
4.5 Results

4.5.1 Exploration and formation of spatial and temporal terms

During the first stage of the study, both agents were able to form consistent maps of the environment, and shared toponymic and duration lexicons were established. The toponymic lexicons for each robot contained seven words and had a coherence of 74.3% (see Figure 4.3). The duration lexicons for each robot contained six words and had a coherence of 80.8%, with durations ranging from 0.3 seconds up to 118.6 seconds (see Figure 4.4).

4.5.2 Use of spatial and temporal concepts

The robots were tested on 25 meet-at games. In all 25 trials, one iRat initiated goal-based navigation within 1-18 seconds of the other at an average of six seconds across all trials. The navigation component proved more difficult than the timing, with 14 / 25 trials successfully completed (both robots found the goal within their duration term length or one robot found the goal and the other established shared attention). Of the unsuccessful games, seven resulted in only one robot finding the goal, and four resulted in both robots failing to find the goal.



(a) (b)

Figure 4.3: Maps and spatial lexicons for the two iRats, a) red iRat's lexicon, and b) green iRat's lexicon.

In the 14 successful games, the robots met each other at an average distance of 0.12 meters and one robot waited for the other robot for an average of 8.6 seconds (see Table 4.1). The distances between the robots at the goal locations varied between 0.05 and 0.28 meters. The waiting times varied between 0.26 seconds and 24.5 seconds. For the toponyms, all of the games except one resulted in the robots meeting each other within the distance used to establish shared attention (0.25 meters). For the duration words, all except two games resulted in waiting times of less than 20 seconds. All times between leaving were under seven seconds.

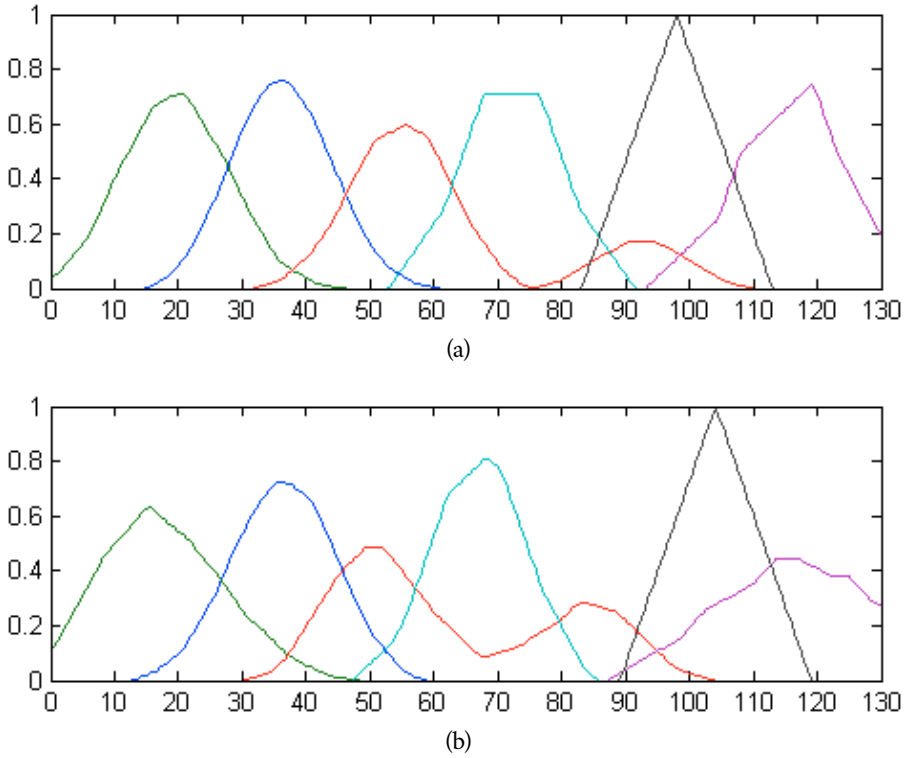


Figure 4.4: Duration lexicons for the two iRats, a) red iRat's lexicon, and b) green iRat's lexicon.

Table 4.1: Results for the 14 successful *meet-at* trials

#	Term	Leaving (s)		Time between (s)		Arriving (m)
		Red	Blue	Leaving	Arriving	Dist
1	fisa	166.02	172.33	6.31	0.26	0.28
2	puni	51.91	55.33	3.42	5.90	0.09
3	puni	51.29	55.33	4.04	24.49	0.12
4	tofe	90.08	84.02	6.06	3.84	0.08
5	puni	51.29	55.37	4.08	9.50	0.18
6	kafi	0.10	2.28	2.18	3.30	0.05
7	fohu	19.65	20.07	0.42	20.84	0.11
8	fisa	165.98	172.33	6.35	6.73	0.08
9	fohu	19.62	20.10	0.48	8.29	0.12
10	fohu	19.63	20.09	0.46	1.03	0.18
11	fedi	37.20	38.98	1.78	16.08	0.17
12	tofe	90.11	84.03	6.08	6.61	0.12
13	tofe	90.12	84.04	6.08	13.81	0.21
14	kafi	0.21	2.29	2.08	0.97	0.07

4.6 Discussion

In extending the Lingodroid methodology from spatial terms to temporal ones, we found many similarities between the spatial and temporal domains. For the distributed lexicon table and associated methods to be useful in constructing and using a relational lexicon, the concept domain needs to be one in which the difference between concept elements can be given a value, such as a distance between locations or a difference in time. The methodology developed for the distance lexicons could therefore be directly ported to the temporal domain of duration.

However, as expected from natural languages, spatial and temporal domains do have some significant conceptual differences. In particular, different methods of linking concept elements to their meanings were required in the underlying representations. Spatial concepts rely on spatial representations that are constructed as the cognitive map is formed, and hence can only be as coherent as their underlying maps. An additional constraint on the specificity of the spatial concepts is the ability of the agents to establish shared attention.

The concept of duration, in contrast, relies on the underlying representation of clock time, with reference to when events occurred in the past. In the current implementation, the knowledge of when events occurred is easily accessible and the clock times are accurate, allowing the formation of a highly coherent duration lexicon. The success of the temporal lexicons belies one of the most intriguing issues of experience. Unlike space, there is an inbuilt arrow of time providing an asymmetry in temporal experiences from the past to the future. The Lingodroids learn words for time based on their past shared experiences and immediately generalize their use to future events.

The practical use of the spatial and temporal terms is demonstrated by meeting at a specified remote location at a specified future time. Predicting the meeting times for future events proved to be the easier part of the *meet-at* task. The effective use of spatial terms has been shown in previous work, but proved more difficult on the new platform.

In porting to a new robot platform (from Pioneers to the iRat), several changes were made that impact on the maps constructed by the robots and on the features that could be used as underlying representations for concept elements. The major issue that affected map quality was that the camera used to obtain views was omni-directional on the Pioneers but forward facing on the iRats. In forward facing cameras, the views for moving one direction along a path do not match the views for moving along the opposite direction, and so fewer connections are made between experiences in the robot's map. When a second robot is also in the same environment, the view of the world is altered when one robot can see the other robot, compounding the challenges for map building. The combination of these factors caused the experience maps formed in the UQ maze to be less coherent than previous studies, which in turn impacted on the coherence of the toponymic lexicons.

Less coherent maps impact on the meet-at game not only in where the robots meet, but also in their ability to find the goal locations in a timely manner. To meet at a particular time, the robots were using their maps to estimate how long it would take to reach the goal location. When the robots took longer to reach the goal than predicted by the map, the waiting time at the goal was affected.

4.7 Conclusions

This work has demonstrated that the Lingodroids can learn terms for durations that are grounded in their own experiences and that can be used in a practical meeting task. We conclude that robots can learn terms for durations in a similar manner to the way in which they can learn terms for distances and directions.

The changes to the cognitive architecture that were needed to make this happen were the additions of a distributed lexicon table for durations together with associated methods for calculating the amount of time since the last meeting of the robots and for calculating the 'temporal distance' between two durations. The additional social interactions were conversations for which an aspect of time, duration in this case, was the topic. There are a variety of ways in which duration can be specified, including the length of a journey, or the time since an event. The conversation in this chapter, *how-long-since-we-last-met*, was a simple way to refer to duration.

The design of the Lingodroid cognitive architecture allowed generalization from spatial to temporal concepts to be achieved relatively easily, and is an indication that other feature spaces may also lend themselves to Lingodroid methods. Generalization to any other space can be achieved by establishing a way to calculate the 'distance' between features in the space, enabling the Lingodroid methods to be used to name relational values based on those features.

4.8 Future work

The full variety of temporal concepts that exist in natural languages extends beyond durations to points in time and the sequencing of events in time. In future work we intend to extend the ability of the Lingodroids to refer to different temporal concepts. The first extension is the use of temporal terms in different contexts and scales. Approximate durations such as 'soon' have a different meaning when a lecture will start (in 10 minutes) compared to when a paper is due (in three days). Given a set of different contexts, it would be useful for a set of terms to be used appropriately across a variety of contexts using a scaling mechanism based on the natural scale of the context itself. Another extension is to name concepts for points in time, which we call temponyms. The interesting thing about points in time is that they can be a specific point that will never occur again, or they can be cyclic, with 'noon' occurring every day and 'spring' occurring every year.

In previous work we have extended experienced distances in a generative manner to be able to learn about and refer to distances that have never been directly experienced. We plan to extend temporal terms in a similar generative fashion. A full set of temporal concepts would enable robots to interact effectively with humans in a wide range of contexts, enabling both discussion about past events and planning of future events.

4.A Description of results

The results described in this chapter follow the lexicon styles for previous Lingodroids studies. Lexicons presented are visualized as per the methodology described in Section 3.2.3. For Figure 4.3, each region represents a different word that is learned by the Lingodroids, and the robots' maps are imposed on top of the figure using blue dots. For Figure 4.4, each differently colored curve represents a duration term and the confidence of association of that term with a range of durations.

Table 4.1 summarizes the results of using the *meet-at* game with the learned spatial and temporal lexicons. The table has 14 rows corresponding to the 14 successful trials within the study. Each row contains the temporal term chosen for the *meet-at* game ("Term"), the times that each robot began moving towards the goal relative to the meeting ("Leaving"), the differences in seconds ("Time between") that each robot began moving towards the goal ("Leaving") and arrived at the goal ("Arriving"), and finally the difference in meters between the robots at the goal ("Arriving Dist"). The table shows that the differences are low between leaving times, arriving times and distances for both Lingodroids.

Chapter 5 – Study II

Long summer days: Grounded learning of words for the uneven cycles of real world events

---

Duration terms allowed the L2 robots to specify temporal tasks, but there are other forms of time that i) are better suited to different tasks, and ii) robots need to be able to learn to understand natural language. Another expression of time that has not been studied before is that of *time of day* (see section 2.3.3). The terms "morning", "afternoon", "evening", "noon" and "night" are commonly used in English to refer to specific times within a day. These terms are particularly suited to expressing events that are defined relative to a day rather than grounded within clock time.

The studies described in this chapter explored grounding time in the uneven cycles of days in a year. "Uneven cycles" refers to the pattern of words for time that repeat every day (e.g. morning, afternoon, night time); however, with uneven lengths across the course of a year (i.e. corresponding to nights that are shorter in summer and longer in winter). Like Study I (Chapter 4), this study contributed to the ability of the L2 robots to communicate about space and time. However, unlike Study I, the grounding used by the robots was not in cognition, but instead in an external, cyclic brightness signal. The motivation behind this study was to extend the L2 robots' temporal cognition beyond clock time to a challenging benchmark task for grounded lexicon learning.

Two studies were designed to investigate how mobile robots could learn and test terms for time of day. In the first study, the robots developed time of day lexicons alongside spatial lexicons and tested the coherence of the time of day lexicons. In the second study, the robots tested their time of day and spatial lexicons using the *meet-at* conversation introduced in Chapter 4.

A new conversation was designed, *what-time-of-day-is-it* to allow the L2 robots to share and ground time of day symbols within lexicons. The grounding used for times of day were formed from concept elements that were each specific instances of a time of day. Each concept element was represented by the tuple $<$ brightness_level, derivative $>$. The *meet-at* conversation was then used to test the time of day terms and the spatial terms that were developed simultaneously.

Like Chapter 4, the L2 robots in this study were again embodied as iRats and the UQ maze environment was used. In the first study, the iRats were allowed to move autonomously around the environment

following the same exploration algorithm from Section 3.1.1 and initiating *what-time-of-day-is-it* and *where-are-we* conversations whenever they had shared attention. The resulting time of day lexicons were presented visually (see Section 3.2.3 for some representative lexicons and how they are visualized). The time of day lexicons from each robot were compared and demonstrated a very high coherence (see Figure 5.6 and 5.9).

One of the issues with using time of day is that the lengths of the day change over the course of a year, and this causes the time of day terms to refer to different clock times during different seasons, particularly for terms like "dawn" and "dusk". This difference from clock time can be an advantage for specifying events that should happen relative to sunlight level. To investigate the effect of day length on the robots' learning, the study was divided into 8 "iRat days" that were 8 minutes long each. The 8 iRat days together formed one "iRat year" that was 64 minutes. The brightness signal and derivative that were provided to the iRats reflected the scaling of days in a year by changing the ratios of day to night within each of the 8 iRat days. Results demonstrate that coherence of the time of day lexicons is very high across the eight equally spaced periods throughout the year.

In the second study, the iRats again moved autonomously around the environment, this time holding *meet-at* games whenever shared attention was established. Ten *meet-at* games using the time of day terms (instead of durations) were held, with seven of the trials ending in success, and another two of the trials failing because one robot blocked the other from its goal (the robots essentially still completed the task in these two trials). Distances between the robots in time and space at the end of each trial are listed.

The conclusions of these studies are again that the L2 framework can be applied to times of day and the associated referents without changing the core algorithms. The use of brightness level and its derivative were found to be sufficient for learning, although neither would provide enough information to discriminate time of day alone.

These studies also provided insight into the nature of time and temporal cognition. Due to the transience of time, when a temponym (a point in time) is named, the term must be generalized to the future in order to be usable again. In Study I (Chapter 4), the use of terms grounded in durations allow reuse by encoding the time differences. Similarly, terms grounded in cyclic events will be re-usable when the event occurs again. Finally, when learning terms for time, some types of time, such as times of day, can only be grounded in real world events (external signals) instead of clock time or other cognition. The learning of different types of time has implications for human-robot interactions, in which robots need to understand all of the different expressions used for time in natural languages to be able to completely understand humans.

The following sections have been reproduced from:

- Heath, S., Schulz, R., Ball, D. and Wiles, J. (2012). Long summer days: Grounded learning of words for the uneven cycles of real world events. *IEEE Transactions on Autonomous Mental Development*, 4(3):192-203

The sections have been taken from the final submitted manuscript and reformatted to fit within this thesis. Note: within this publication, the L2 framework is referred to as just Lingodroids.

Abstract – Time and space are fundamental to human language and embodied cognition. In our early work we investigated how Lingodroids, robots with the ability to build their own maps, could evolve their own geopersonal spatial language. In subsequent studies we extended the framework developed for learning spatial concepts and words to learning temporal intervals. This chapter considers a new aspect of time, the naming of concepts like morning, afternoon, dawn, and dusk, which are events that are part of day-night cycles, but are not defined by specific time points on a clock. Grounding of such terms refers to events and features of the diurnal cycle, such as light levels. We studied event-based time in which robots experienced day-night cycles that varied with the seasons throughout a year. Then we used *meet-at* tasks to demonstrate that the words learned were grounded, where the times to meet were morning and afternoon, rather than specific clock times. The studies show how words and concepts for a novel aspect of cyclic time can be grounded through experience with events rather than by times as measured by clocks or calendars.

5.1 Introduction: Beyond clock time

What is meant by a word like "dawn", which is both an event and a time of day, is an intriguing question. How can a robot gain a grounded sense of that meaning? As the year cycles through the seasons, midday and midnight keep their positions as sentinels of peak dark and light, but dawn and dusk can move by several hours, occurring at different points on the cycle.

One useful perspective on a word and its meaning is provided by Peirce's semiotic triangle (Ogden and Richards, 1923), which ascribes a referent in the world and an internal representation to each word. Robot language learning has previously been developed within a framework adapted from Peirce (Vogt, 2002; Steels, 2005; Roy, 2005; Schulz et al., 2011a, 2011b). Harnad (1990) highlighted the challenging nature of the relationship between a word and its meaning, calling it the symbol grounding problem. Words for some physical objects can be grounded in sensorimotor percepts (Vogt, 2002; Steels, 2005; Roy, 2005; Steels, 2008; Schulz et al., 2011a, 2011b), while other concepts may be grounded in secondary structures (Cangelosi and Riga, 2006; Cangelosi et al., 2010; Schulz et al., 2011a, 2011b; Uno et al., 2011).

However, the semiotic triangle and current solutions to the symbol grounding problem in robot language research assume that, for each symbol, there is a corresponding internal representation in the

human or robot brain, and also a referent in the world to which the symbol can refer.

Is there a referent in the world to which the word dawn corresponds? Dawn differs from typically discussed grounded terms (like a table) in that it is related to an event, not an object. Dawn is the time of day when the sun rises. It is also unusual in that the word can be used to refer to a time (e.g. meet at dawn), but its defining features, the rising sun and changing light levels in the world, are not themselves temporal.

We argue in this chapter that words for the uneven periods in the cycles of real-world events (like dawn, dusk, morning, and afternoon) provide challenging benchmark tasks for grounded robot language learning. The development of processes that enable robots to learn the grounded meaning of words for cycles in time leads to consideration of fundamental issues about time and timing that are not typically addressed in robot–human communication. Our goals, in this chapter, are to propose the grounded learning of terms in cyclic time as a benchmark task, present a solution using Lingodroids, and then analyze the solution and what was needed to accomplish the task.

5.1.1 Robots, clock time, and grounded language

Time is fundamental to human language and both the concepts and names for temporal terms will form an essential part of an agent's embodied knowledge and grounded language. A robot will need to learn many different temporal concepts to understand human language. The challenge for designing a robot's cognitive architecture is how much needs to be, or even can be, encoded *a priori*, and how much needs to be learned through embodied action.

The simplest concepts of time relate to the clock and calendar, as an orderly progression of time points that can be used by agents to synchronize their actions, either with humans, other agents, or events in their environment. Clock time is useful for ensuring events happen at particular times with high temporal precision. We call particular times and dates temponyms, analogous to the naming of particular places, which are known as toponyms. Temponyms assign unique referents to temporal points, such as midnight on December 31, 2000, and constitute landmarks in time. Although humans are not born with built-in clocks and calendars, the concept of temponyms can be easily represented by robots, and representations based on accurate clocks seem relatively straightforward.

However, time in natural language is more complex than clock and calendar time (Sinha et al., 2011). Relatively few agent-based studies have learned and named temporal concepts (Steels and Baillie, 2003; De Beule, 2006).

The evolution of lexicons for terms denoting the sequencing of events was studied by De Beule (2006), who used discrimination games to form an ontology of time terms based on the evolution of language approach of Steels (Steels, 2005). de Beule used software agents (rather than embodied robots) and assumed that the agents had access to high-level representations of events as predicates, such as $fall(X)$ and $past(X)$, and could form sequencing of past and present when required.

Even before the formation of predicates, there are open questions for grounding temporal concepts referring to durations. How long is a *short journey*? When is *soon*? Many references to durations use approximations learned in context. There are also deep questions about how time itself is understood. Many metaphors map time into space, such as time as a "path" and life as a "journey" (Clark, 1973; Boroditsky, 2000; Gentner, 2001; Gentner et al., 2002). To examine spatial metaphors for mapping time

into space, in previous studies we used Lingodroids, language learning robots with the ability to explore and map their environment (Schulz et al., 2011a; Schulz et al., 2011). Like de Beule, our first studies to add temporal terms to the Lingodroid lexicons were done in simulation (Schulz et al., 2011b). The bio-inspired mapping system, RatSLAM (Milford and Wyeth, 2010), was used to create maps in a virtual reality world. We studied how words for temporal durations could be learned by estimating the lengths of shared journeys through that world. The studies used conversations around the question "*How long did it take?*" The grounded meaning for terms was compared in an extensive set of comparisons ($3 \times 2 \times 3$ study design). Three different factors were used to specify the length of a journey: i) distance traveled; ii) the time of the journey calculated as the sum of the times taken for each segment; and iii) the amount of change experienced along a route. Two different memory conditions were used. One was based on a specific journey (instances) and the second one was based on the shortest path from memory between two known locations (prototypes). Three lexicon sizes were created, resulting in small (1–5 words), medium (3–10 words), and large (6–19 words) lexicons. The studies showed that all duration concepts (distance, time, and change) resulted in useful grounded lexicons, but as expected, durations based on specific instances of journeys enabled the agents to evolve the most coherent lexicons.

Translating simulation studies to real robots provides many and varied challenges. In particular, there are frequently unexpected issues as the world has sources of noise and change that are controlled or simply not present in simulation worlds. The Lingodroid studies of time durations were modified to be practical in a real-world setting based on conversations such as "*How long since we met?*" and implemented on a new robot platform, called the iRat, which is a rat-sized robot developed for studies at the intersection of neuroscience, bio-robotics, and embodied cognition (Ball et al., 2010). The spatial mapping system, RatSLAM (Milford and Wyeth, 2010), was ported to the iRat from the Pioneer 3 DX robots used in earlier spatial language studies (Schulz et al., 2011a; Schulz et al., 2011). Although running the same algorithms, the iRats have a different physical embodiment to Pioneer robots. The most obvious difference is their size. The iRats are much smaller which enables them to be used in relatively complex but compact environments. Their size also means that their perspective is much closer to the ground, changing the way they interact with objects around them. The major difference for the *how-long-since-we-met* studies turned out to be the cameras. The iRats were developed with forward-facing cameras rather than omni-directional ones, and this difference impacts the way paths are connected to create maps in RatSLAM. Using an omni-directional camera, moving north or south along the same path can be equated through a visual transform (i.e., in software). Using forward-facing cameras, no such visual transform is possible. Despite the increased difficulties in creating effective maps, studies showed the iRats did create effective lexicons of spatial words to describe their arena and temporal terms to describe their previous meeting times. Duration words were formed corresponding to approximate durations from 0–130 seconds. Spatial terms were created for different areas of the arena.

One of the issues with real robots that can be finessed in simulations is the establishment of shared attention to a particular topic. In the Lingodroid studies, it was important to consider how the robots would establish attention to the points in time and space that they would consider "now" and "here". The robots cannot both occupy exactly the same point, so when they discuss locations (called *where-are-we* conversations), they always have slightly different positions on a map. Since one robot speaks first, there

are also small differences between their shared attention to the "current" time which is slightly later for the listening robot than the speaking robot. In practice, shared attention to points in time was less of an issue than points in space.

Evaluating the success of evolution of language studies is not a widely discussed issue, but it is critical to the future of human-robot interactions. With many evolution of language studies, the studies are evaluated by the coherence of the language for all members of the community. That is, success is assumed if it is shown that words used by different members of a community refer to the same concepts. This measure does not guarantee that the concepts and their words are useful in practical tasks.

With fully embodied robots, it is possible to demonstrate that the words are fully grounded in the real world by testing their use in novel conversations that require actions by the robots. The Lingodroid temporal lexicons from the *how-long-since-we-met* studies were tested using a series of *meet-at* tasks in which one robot specified a location and future time, and both robots attempted to meet at the specified location and time (Heath et al., 2012a). Despite high coherence between the robots' lexicons, *meet-at* proved a challenging task. Out of 25 trials where the robots attempted to meet at a specified location at a specified time, 14 were fully successful, with the robots meeting at the right place within a close period of time. Analysis of all trials showed that the robots' intended temporal meeting times were accurate in all cases, but the spatial navigation abilities limited the robots' success in reaching the target locations within the specified times.

All the studies to date that evolve lexicons for temporal terms have ultimately been grounded either in clock time (measured times when events such as journeys started and finished (Schulz et al., 2011b)), metaphors for time (such as the durations and amount of change in the Lingodroid *how-long-did-it-take* studies (Schulz et al., 2011b)) or sequences (Steels, 2005; De Beule, 2006). Although words like soon were approximated from experience, the lengths of durations did not change with changing events in the world.

This chapter addresses the learning of a new type of temporal term, *event-based time*, that cannot be measured by clocks, but rather changes with the changing patterns of the environment.

5.1.2 Cyclic time: Changing day lengths from Summer to Winter

As noted above, a day-night cycle is 24 hours, but critical points, such as dawn and dusk, refer to events in the world, as the sun rises and sets. On New Year's Day in Brisbane, the sun rises at 4:55 AM. Six months later the sun will rise nearly two hours later at 6:39 AM. Further from the equator, the day lengths are much more extreme. Human language incorporates important concepts that are tied to cyclic events, but not to the clock. To study how robots could agree on words for the parts of a day (morning, the middle of the day, afternoon, and night) a variety of questions need to be addressed:

- What social interactions will enable robots to learn a grounded lexicon for human-like temporal concepts?
- What representations are needed in the robots to represent cyclic time concepts?
- What environmental conditions should the robots be exposed to?

5.2 Methods for grounding cyclic times on sunlight levels

5.2.1 A signal for cyclic time

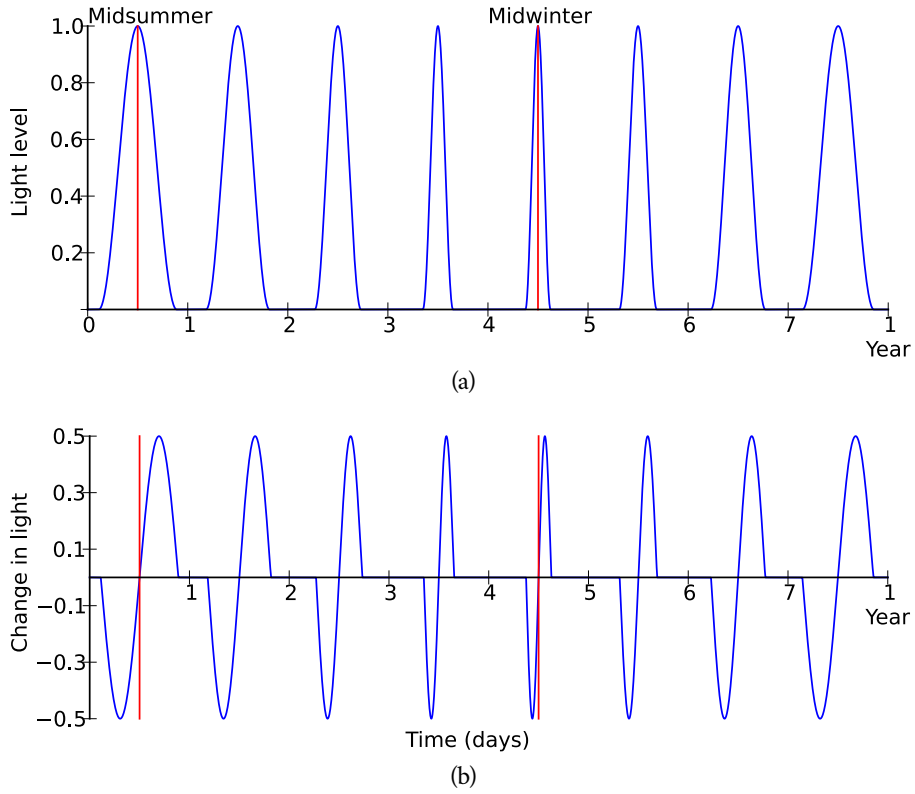


Figure 5.1: Sunlight levels in the iRat year. Each iRat day is eight minutes long and there are eight days per year (3840 seconds in total). (a) Sunlight levels (y-axis) over one iRat year (x-axis). (b) Derivative of the light levels for the same period.

For agents to be able to express cyclic time, the vocabulary must be grounded in cyclic features, such as days and years. To address this requirement, a simulated sunlight level is provided to the two agents as if sunlight can stream through the window in the lab.

An iRat day is defined to be eight minutes (480 seconds) long, and an iRat year is eight days (3840 seconds) long. That is, the daily cycle begins at midnight, which is completely dark, progresses through dawn to the brightest levels at midday (four minutes later), then the light levels decrease through afternoon and evening to black again at midnight (the whole cycle taking eight minutes). To show the changing lengths of day, the cycle starts at midsummer (day one) with the longest days and shortest nights, proceeds through the autumn equinox to winter (day five), and then back through the spring equinox to summer again. The sunrise equation (Cornwall et al., 2010) (derived from Meeus (1991)) is used to generate the light signal by calculating the predicted day-to-night ratio for eight days of the standard year at even intervals. This ratio is then applied to an eight-minute duration to calculate how many minutes should be daytime and nighttime. A single period of a sine wave is then scaled to fit the predicted day lengths so that there are changing light levels during the day and a constant light level of zero at night (see Figure 5.1a).

The shortest feasible periods are used for the iRat day and year to balance the advantage of running a full (simulated) year in a single experiment (the iRat's batteries last 1–2 h) while enabling a reasonable number of trials to be run each day. The sunlight signal is simulated rather than changing the lighting for

the entire environment to enable the study to be run in a computer lab. Light levels alone are not sufficient to identify points on the daily cycle. Dawn and dusk can have equivalent levels of light, but occupy quite different times. The change in sunlight is also important. Hence, a signal proportional to the derivative is also provided in the form of a similarly scaled cosine period for each day (see Figure 5.1b).

5.2.2 Language conversations for naming times of day

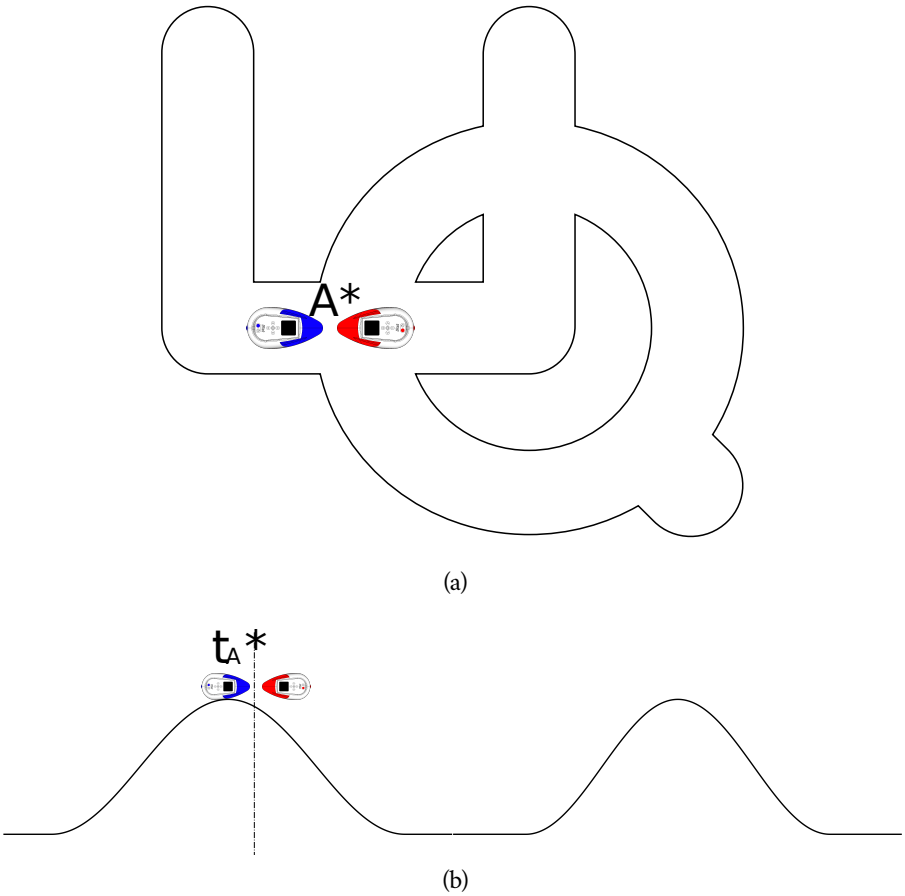


Figure 5.2: Two iRats have shared attention when they are in close proximity. Whenever the iRats are close to each other, they will start a conversation. a) In a *where-are-we* conversation, the topic will refer to the current location of the iRats indicated as A^*. b) In a *what-time-of-day-is-it* conversation, the topic is the current time indicated as ${t_A}^*$, and the iRats pay attention to the light levels and their derivative (shown as just after midday in b). The "$*$" indicates that new terms can be created and that both iRats will associate the chosen term with the feature.

A new type of conversation called *what-time-of-day-is-it* was developed to allow agents to create terms describing times within the day/night cycle (see Figure 5.2).

Each conversation starts with one iRat asking the other for a term to describe a feature. The responder then either suggests a term that has already been used for places close to the feature or invents a new term if there are no suitable candidates.

what-time-of-day-is-it conversations are conducted similarly to previous social interactions such as *where-are-we* and *how-long-is-it* requiring only a single response: a time of day. The iRats do not use clock time and instead rely only on the generated sunlight signal as an input. Like all previous Lingodroid social

interactions for space and time, the problem of shared attention to a feature is solved implicitly by the fact that when two agents meet, their coordinates and clock times are approximately the same, and apply equally well to a shared sunlight signal.

5.2.3 Demonstrating the utility of grounded terms

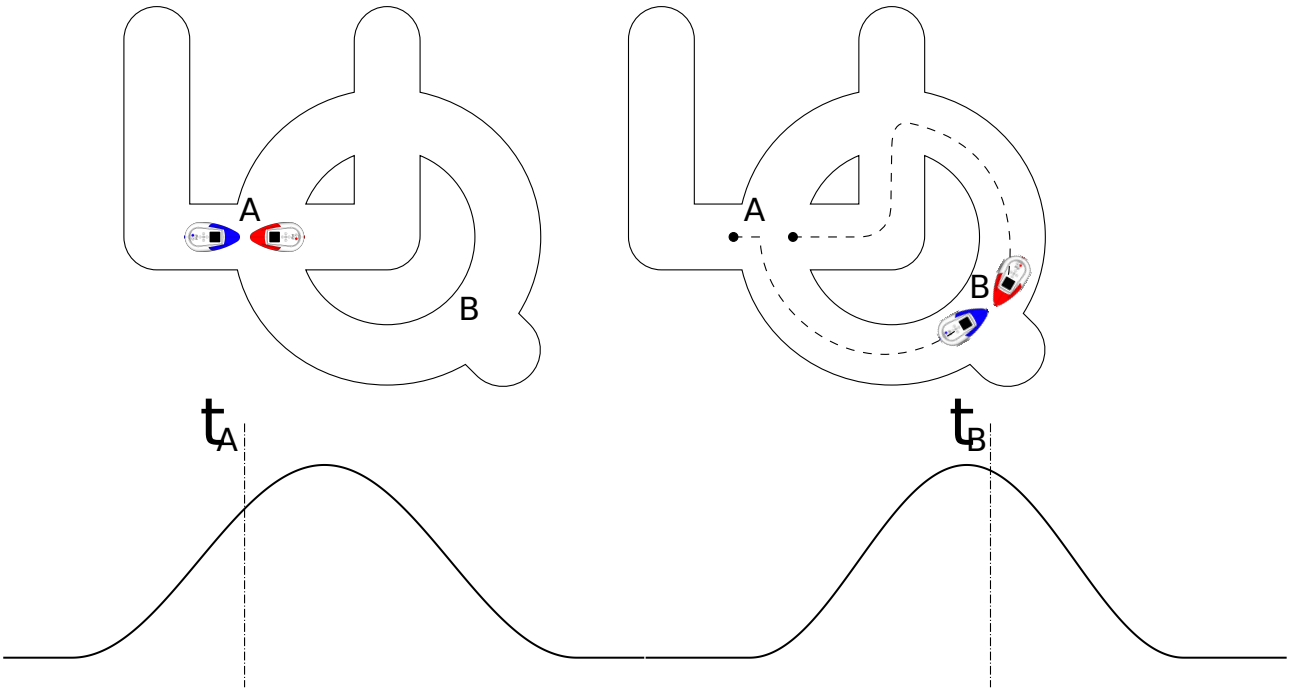


Figure 5.3: The *meet-at* task involves the two iRats establishing shared attention at place A and then a speaker deciding on a time through proximity at time and place B to meet. The robots wait until time and then independently navigate to place B; it is possible for the iRats to move between any two locations on the map within one period of day (such as leaving and arriving within the morning). There is no learning as the task is designed to test previously grounded terms for place and time.

A previously developed task, *meet-at*, was adapted to demonstrate the application of the learned terms to performing a task (see Figure 5.3). The *meet-at* task tests the terms learned from the *what-time-of-day-is-it* conversations and applies them to a meeting task. For *meet-at*, one agent tells the other agent a word for a place and time to meet and then both agents attempt to navigate there. The *meet-at* tasks consist of three phases: i) the initial discussion; ii) a waiting period; and iii) navigation to the goal. In order to generate places to meet, *where-are-we* conversations are used to generate toponyms for places (Schulz et al., 2011a). As the *meet-at* task is used solely for testing shared understanding, no change in the association of terms in the lexicon is performed during this task.

5.2.4 The iRats and their environment

The iRat has the functional capabilities of a PC on wheels. It has a rodent-inspired robot shape that is about the same size and weight as a large rat (see Figure 5.4) (Ball et al., 2010). The robot has IR sensors for avoiding obstacles and a wide-angled forward-facing camera for performing visual SLAM (see Table 5.1 for data ranges for each sensor). The iRat uses Robot Operating System (ROS) as its communication method,

allowing high level controllers to be placed offboard (Quigley et al., 2009). The two iRat controllers are run on an external computer and communicate with the iRats using ROS interfaces. ROS is also used for passing words between the two iRat agents.

Figure 5.4: iRats talking. iRats randomly explore their environment until they meet. When they are within hearing distance of each other, they stop, and have a conversation.

A maze constructed from a "U" and a "Q" (the initials of the University of Queensland) cut out from a 55mm high foam sheet was used as the iRats' environment (see Figure 5.5). The UQ maze was adapted from a previous telerobot study (Heath et al., 2011). The maze was designed so that the interlocking U and Q create three different loops allowing for odometry information from the iRat to be corrected by loop-closure when using visual SLAM. The UQ maze was also designed to fit within a laboratory, covering an area of 2.07m$\times$1.87m. The iRats randomly wander around by employing simple left, right, and center wall following behaviors, switching between different left and right walls with low probability.

The iRats create their maps using RatSLAM, which is a simultaneous localization and mapping (SLAM) system inspired by the structure of the rodent hippocampus (Milford and Wyeth, 2010). The iRats run a minimal version of RatSLAM which supports mapping and goal-based navigation but omits the pruning algorithm used in other implementations for persistent navigation.

5.2.5 Overhead tracking the iRat motions and ground truth

The iRat environment contains a visual overhead tracking system, which can return the ground truth locations of both iRats in pixel values within an image. For all the iRat social interactions, the overhead tracking system is used to determine positions of the two iRats and establish conversations when the distance apart is less than 0.25m.

5.2.6 Grounding and representation of concept elements

As the iRats explore their environment, the RatSLAM algorithm records their journeys as graphs in which each node represents an experience to which sensory inputs are attached, and links represent odometry

Table 5.1: Physical sensors and actuators

Sensors / Actuators	Data Range
Forward facing camera	416x240 RGB images
Three Sharp IR sensors	0.1-0.4m
Wheel encoders	0-0.5m/s and 0-2rad/s
Motors	0-0.5m/s and 0-2rad/s

between those experiences.

A new type of concept element was added to the lexicon table for the current studies to add the sunlight signal. The sensory data attached to each concept element includes a 2-D vector encoding the light level and its derivative. Each $<$ light level, derivative $>$ is considered a unique concept element. The stored information represents the features of the event (not the clock time at which they occur); however, the algorithms interpret the information as a signal about the time of day. This is a critical aspect of the Lingodroid temporal learning system, and has implications for how the robots learn the grounded meanings of words.

Representing the event rather than clock time (i.e., storing time as a vector combining a sense impression and its derivative) has two implications: i) each element uniquely identifies a point of time in a day, and ii) the changing day lengths scale with the seasons throughout the year.

Through their interactions, the iRats need to learn concepts and words to cover all their shared experiences. Whenever they are engaged in a conversation, if their sense vector is not sufficiently close to one already remembered, a new element is created and stored. The resulting elements are distributed over all times of day at which a conversation occurred.

5.2.7 Distributed lexicon table

A key structure in Lingodroids is the distributed lexicon table, which is used to associate words to concept elements with a many-to-many mapping. The data structures within the lexicon include a dictionary of words, an array of concept elements, and a matrix of associations from words to concept elements. The concept elements for the *what-time-of-day-is-it* conversations are the $<$ light level, derivative $>$ vectors that are dynamically added to the lexicon structure every time a new time of day is experienced.

The association a_{ij} between concept element i and word j is incremented whenever word j is used when concept i is active, as follows:

$$a_{ij}{}' = a_{ij} + 1.$$

5.2.8 Word production and comprehension

One of the characteristic features of Lingodroids is the way in which concept elements are grouped into categories denoted by words. In many evolution of language studies, categories are first formed and only subsequently named. In Lingodroids, instances are associated with names as described above, and each word denotes a generalization over that set of instances.

Figure 5.5: The environment used for the iRats language studies. It is created from a U and a Q cut from a foam sheet.

As a consequence, classification performed by the Lingodroid algorithm is an emergent process effected during word production. The advantage of this approach is that the definition of a set of features can remain unknown until it is required in a conversation. Word production was used in both the learning and testing interactions.

The word production and comprehension algorithms have been described in previous Lingodroid papers (Schulz et al., 2011a). They are repeated below for completeness and to show how the concepts are adapted to the lighting concepts and temporal terms.

For the *what-time-of-day-is-it* conversation, the current situation is defined using the $<$ light level, derivative $>$ vector. To choose an appropriate word for an element, the Lingodroid word production algorithm selects the set of concept elements within a given neighborhood of the current element. The confidence h_{ij} that word j should be used with concept element i is calculated as follows:

$$h_{ij} = \frac{\sum_{m=1}^{Y} \frac{a_{mj}\,(D - dist_{mi})}{D}}{\sum_{n=1}^{N} a_{nj}},$$

where D is the neighborhood size, Y is the number of concept elements in the neighborhood of element i, N is the total number of concept elements, and a_{ij} is the association between element i and word j. A neighborhood size of 0.5 m is used for D.

Words are invented with probability p as follows:

$$p = k \exp\left(\frac{-h_{ij}}{(1 - h_{ij})\,T}\right)$$

where $k = 1$, h_{ij} is the confidence value that word j should be used with concept element i, and T is a temperature parameter for scaling the rate of word invention, set to 0.1, which allows a small probability

of creating new words at any stage throughout the learning part of the experiment.

An interesting problem arises if the two robots assign different words to the same set of features. This can happen in two different cases. In case one, two robots each assign a different word to the same sensory category. In case two, a single robot assigns two different words to the same sensory category. Both cases are only possible in the toponym games, caused by the incorrect localization of an agent during a conversation. Case one occurs when the incorrectly localized agent is the listener. In resulting conversations, the word with the highest confidence would be chosen to express a feature. Case two occurs when the agent incorrectly localized is the speaker. In later conversations, both words remain candidates for expressing a feature, but typically one of the words is used more frequently and eventually becomes the symbol of choice for that location. The outcomes of both cases are probabilistic, as they depend both on the choice of the speaker at the next game in that area and the values of words given by previous games around that area.

5.2.9 Measuring grounding success

Grounding success is measured in two ways: i) a coherence test of the entire lexicon, as used by the two iRats, and ii) the performance of the robots on a practical task based on the grounded terms. For the *what-time-of-day-is-it* conversations, the first measure is calculated as the lexicon coherence between the two agents and the second measure as the distance apart in both time and space after the completion of a *meet-at* task. For the latter, the overhead tracking system is used to calculate the ground truth distance between the two agents. The differences in time are calculated using the agents' clock time.

5.3 Study 1 – Grounding day-night terms in *what-time-of-day-is-it* conversations

5.3.1 Aims

The aim of the first study was to explore the grounding of cyclic terms for time using changing light level signals, and evaluate how the grounded terms can scale as the day length changes throughout the year. It was predicted that the terms in the lexicon would automatically scale with the variable length days, affecting both iRats equally and hence maintaining the shared grounding of terms.

5.3.2 Methods

The study involved two iRats that explored their environment by randomly following walls. When they met, they held a conversation about their location (*where-are-we*) and the time (*what-time-of-day-is-it*). After completing the conversation, they returned to exploring. The experiment was continued for the period of one iRat year (64 minutes). There was no human intervention involved throughout the conversations. The iRats would autonomously end one conversation and continue exploring before meeting again for the next one. The resulting lexicons were saved to files and images and were post-analyzed by calculating the confidence and the production values of words for each possible time of day during the course of a year.

5.3.3 Results

During one iRat year, a total of 58 *what-time-of-day-is-it* conversations were held. Each lexicon generated four words on average, corresponding to different light levels throughout the day (see Figure 5.6). One term referred to the darkest period, corresponding to night, and covered the longest period, with the other three terms distributed throughout the daylight period. The three daylight words evenly spanned the daytime.

The lexicons created by the two iRats were very similar, as can be seen in the production value pie charts (see Figure 5.6). Lexicon coherence was calculated as the pixel difference between the production values (see right hand image in Figure 5.6). The predominance of white indicates that the iRats agree on

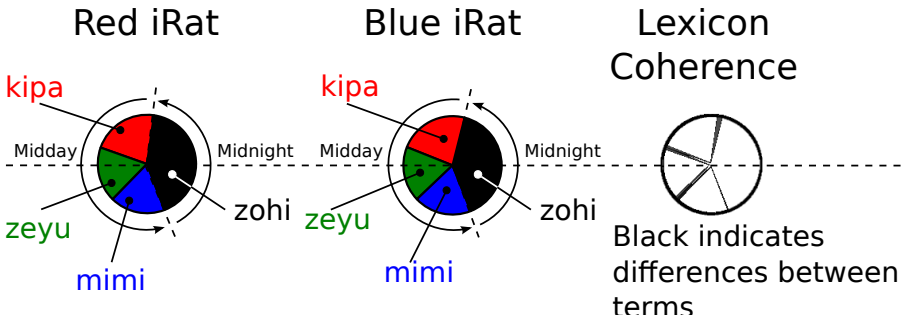


Figure 5.6: The two iRats production values for the longest day in the year. Note that midnight starts at 0° pointing right and then the day cycle goes around the circle anti-clockwise as in Cartesian coordinates. The differences between lexicons are calculated as a pixel difference of the two pie-graphs (midday and midnight are not learned terms, and are shown for clarity).

the prototypical use of the terms, with differences only seen in their choice of terms when moving from one time period to the next.

The words generated by the two agents for the different times of day are shown, with their production values, in Figure 5.7. Production values are the words that are used in communication to describe a feature and although the word confidence values may differ visually, the production values are almost identical with only the slightest shifts of less than five seconds between each term boundary.

Confidence values were calculated for each time of day for each word (see Figure 5.7). The two robots' lexicons divided the day into similar temporal regions, with slight differences in the confidence values of words, seen in the term for morning (kipa).

One of the essential issues in time-of-day studies is that the day lengths change throughout the year. Confidence values were calculated for the full year (see Figure 5.8 for values over half a year). Because words are grounded in sunlight and not in clock time, as the sunlight signal changes, the word use expands and contracts.

Production values for a full year demonstrate how the production of terms scale throughout the year (see Figure 5.9). This scaling of events requires no continuous updating of lexicons by the robots but purely the choice of events in which to ground symbols for time provides the automatic scaling of the lexicons for each part of the year. In the robots' cognition, a symbol is always linked to the same light level and derivative pair; however, the pairs are not always linked to the same time of day.

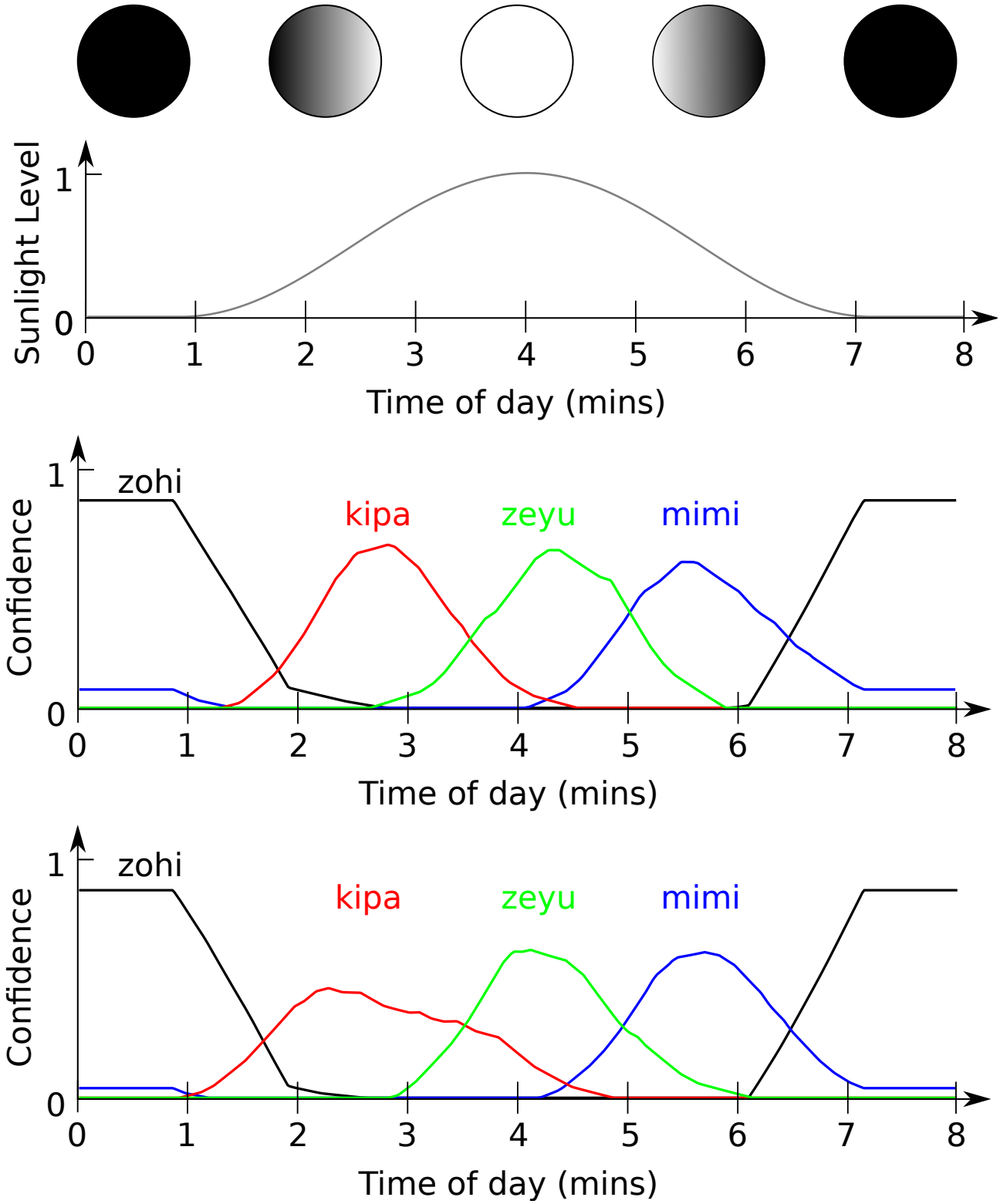


Figure 5.7: Time of day lexicons. The *what-time-of-day-is-it* conversations generated lexicons containing four words. A typical lexicon is shown. a) The features that are used as inputs for the times of day are based on light level and change in light level. b) The light level over the course of the day. c) Red iRat's temporal lexicon. d) Blue iRat's temporal lexicon. Both iRats have divided the time in a day using three descriptive words that approximate morning ("kipa", red line), midday ("zeyu", green line), and afternoon ("mimi", blue line) and one word for night ("zohi", black line).

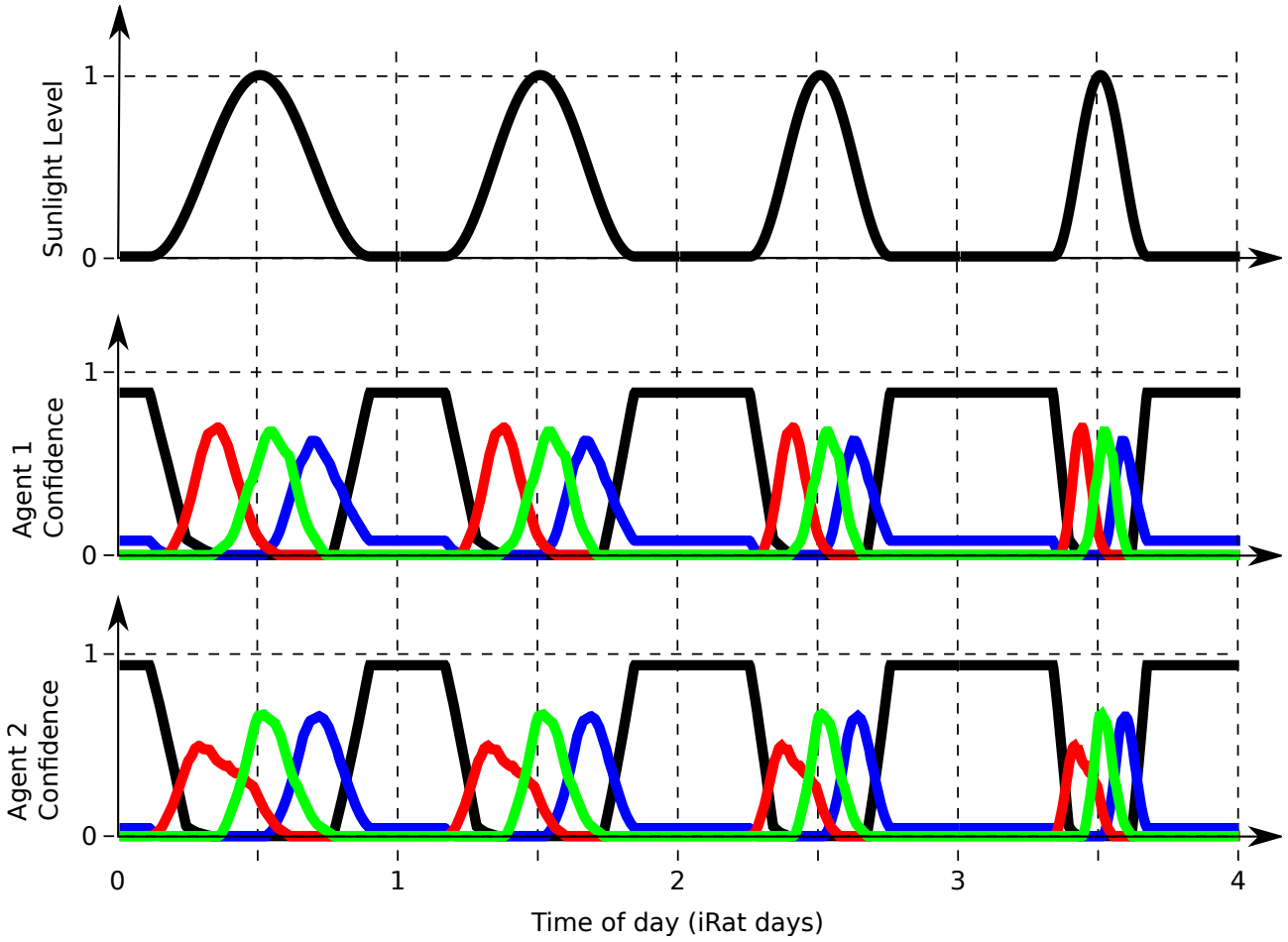


Figure 5.8: Word use for times of day scale to the different day lengths from summer to winter. a) The light levels over a half year (four day) period from long summer days to short winter days. b) the red iRat's lexicon. c) the blue iRat's lexicon. Word use scales with the light levels.

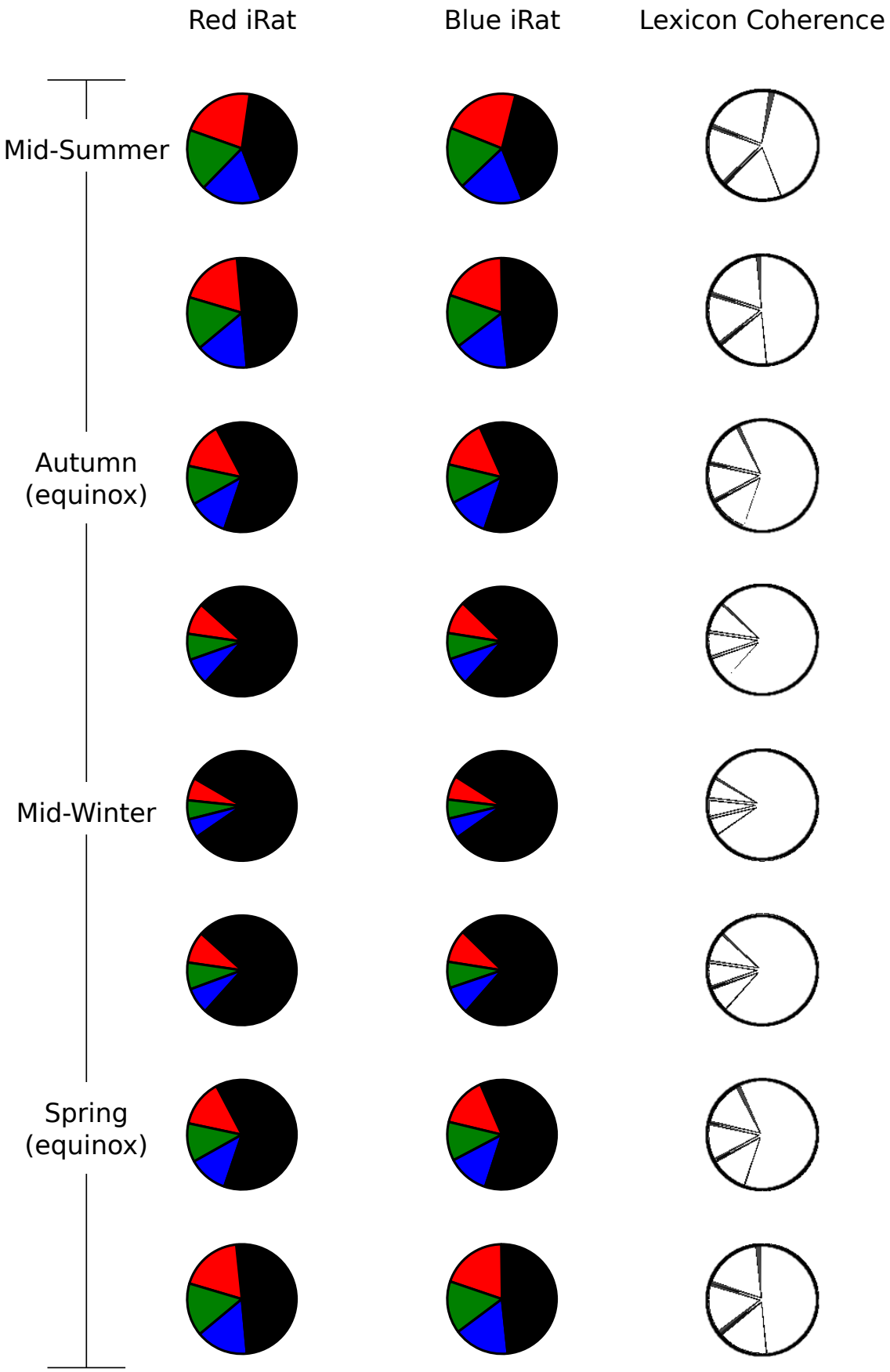


Figure 5.9: Word use and coherence over the iRat's eight-day year. The circles in this figure show the production values used for the time of day through a full day-night cycle for each day of the iRat year. Each row shows a different length day and night corresponding to the eight days in the iRat year along with lexicon coherence.

5.4 Study 2 – Evaluation of grounded meanings using *meet-at* tasks

5.4.1 Aims

The aim of the second study was to test the effectiveness of using a vocabulary grounded in cyclic signals to perform meeting tasks. In the earlier simulation studies of Schulz et al. (2011b), agents in *meet-at* tasks would correctly estimate times, but had less success with their map coherence. We predicted in these studies that the robots would have a clear idea of what time they should meet, but it was an open question as to how difficult they would find it to reach the meeting place within the time chosen.

5.4.2 Methods

The lexicons and RatSLAM maps from the *where-are-we* and *what-time-of-day-is-it* conversations held in Section 5.3 were used as a starting point for study 2. The two iRats explored their environment randomly and whenever they met, a *meet-at* task was started. Exploration continued until a total of ten *meet-at* tasks were attempted. Results were collected on the words used, times chosen, and the difference between arrival times and distances. Overhead tracking was used to record the robots' positions at all times. The term for night (*zohi*) was deliberately omitted from this experiment as the length of the night was so much longer than the lengths of the daytime terms, scheduling a meeting time using the term for night would likely succeed even when the agents arrive at quite different clock times.

5.4.3 Results

Out of the ten *meet-at* tasks attempted by the iRats, seven were completed successfully, with both robots reaching the meeting location at the agreed time (see Table 5.2). The three trials that failed were the result of one of the iRats being unable to navigate successfully to the goal. Two of these failures were caused by one iRat blocking the other iRat from its desired place. Navigating around an obstacle in narrow paths like the ones used in these experiments is a difficult problem. The third failure was caused by the navigation system failing and the red iRat becoming stuck in a repeated loop. For the three failures, times and distances were recorded when the iRat gave up. For trials two, seven, and nine, the give-up times for the red iRat were 58.76, 68.7, and 40.3s, respectively, indicating that the iRat had chosen the correct time to start moving. Their corresponding separation distances at the time when the iRat gave up for trials 2, 7, and 9, were 0.8, 0.2, and 0.5m, respectively. For the latter two, the iRats actually touched at some point during the correct time period but because the red iRat was incapable of reaching the goal it had set for itself, the result was still counted as a failure. The timeline of a successful meet-at trial was examined in detail (see Figure 5.10). The term for middle of the day (zeyu) was chosen as the target meeting time in this trial. There were only two seconds between the times of day that the robots started moving to the goal. One robot took 27 seconds and the other 38 seconds to reach their target locations. It can be seen from this example that there is far more variability in the time taken to navigate to a goal than variability in time differences between terms in the lexicon.

Table 5.2: Results for 10 *meet-at* trials

Trial	Time	Arrived		Between Arrivals		Goal
#	Chosen	Red	Blue	Time (s)	Dist (m)	Success
1	kipa	kipa	kipa	12.20	0.30	Yes
2	mimi	-	mimi	-	-	No
3	mimi	mimi	mimi	1.15	0.40	Yes
4	kipa	kipa	kipa	2.60	0.30	Yes
5	zeyu	zeyu	zeyu	10.90	0.50	Yes
6	kipa	kipa	kipa	21.60	0.50	Yes
7	mimi	-	mimi	-	-	No∗
8	mimi	mimi	mimi	4.74	0.60	Yes
9	mimi	-	mimi	-	-	No∗
10	zeyu	zyu	zeyu	30.85	0.30	Yes

∗ indicates that trial was unsuccessful because the iRat that failed to arrive timed out attempting to get to a goal on the other side of the iRat that had already arrived.

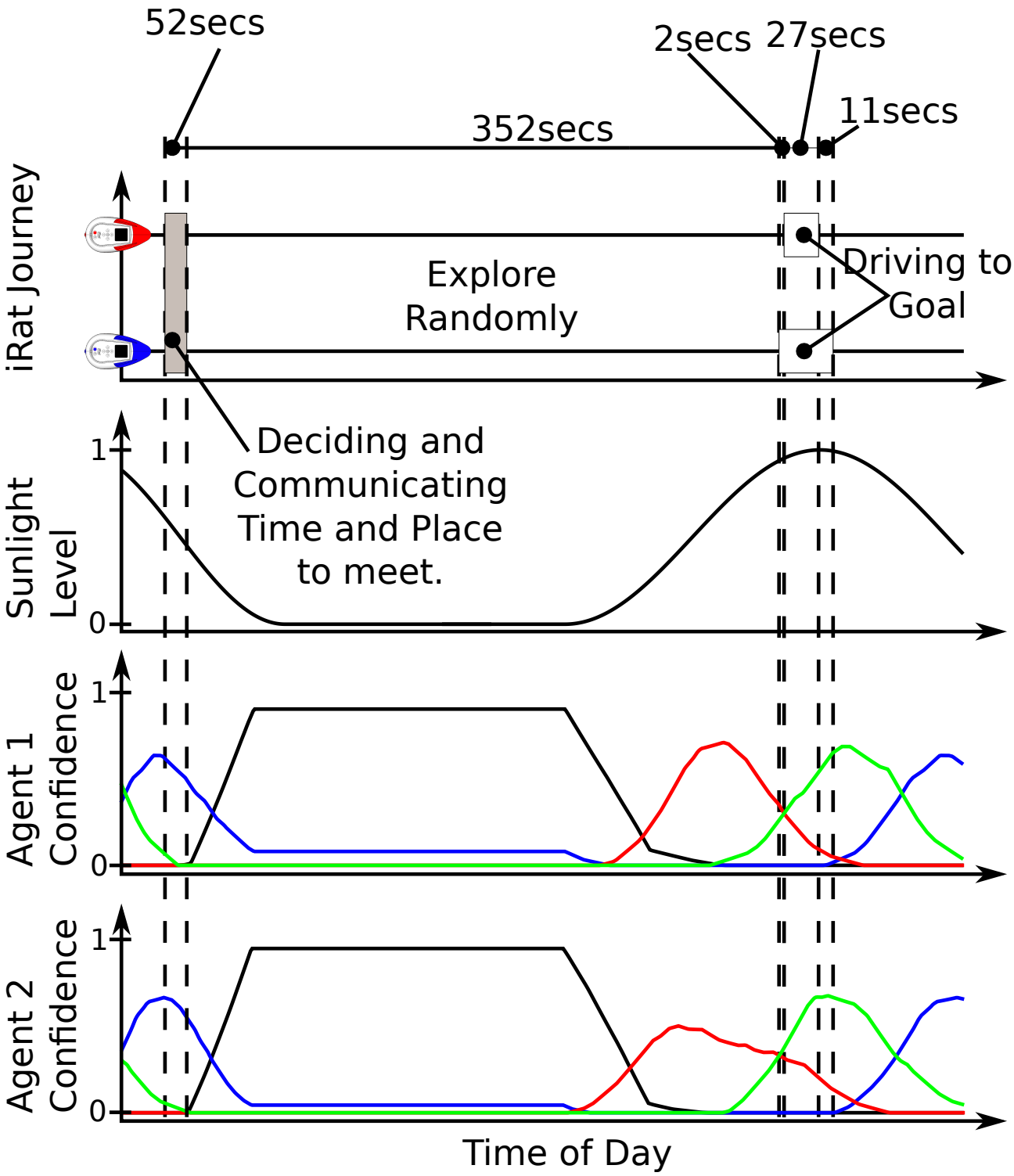


Figure 5.10: A successfully completed *meet-at* task corresponding to trial five in table 5.2. The robots met at an initial time (shown as the vertical dashed line at the left of the figure) and then chose a time and place to meet again using words from their lexicons. For this trial the time chosen was zeyu (shown in green). Because of the slight differences in lexicons, the red robot started driving two seconds after the blue robot. The blue robot spent more time driving to the goal and the red robot eventually arrived before the blue robot by 11 seconds.

5.5 Discussion

The studies presented in this chapter demonstrate how robots can learn words for temporal terms grounded in events in the world, such as the levels of sunlight. They also demonstrate that the robots have developed grounded meanings from sunlight levels that they can use in practical tasks to meet at particular times.

Event-based temporal terms occur in natural language (like night, day, dawn and dusk), but clock time dominates the majority of studies of time. The studies in this chapter demonstrate the ease with which an event-based foundation for temporal concepts can be developed. Linguists have studied a wide range of cultures to examine how time is understood and spoken about (Levinson, 1996; Núñez and Sweetser, 2006; Evans, 2010). In cognitive linguistics, metaphors for time have been emphasized (Lakoff and Johnson, 1980), particularly mappings from time into space (Clark, 1973; Boroditsky, 2000; Gentner, 2001). However, this view has been challenged recently, with an increasing understanding of the diversity of the languages and cultures of the world. One interesting case is Amondawa, the language of a people from Western Amazonia whose traditional way of life is dominated by hunting, fishing, gathering, and small-scale cultivation. A recent report by Sinha et al. (2011) emphasizes the contrast between Amondawa's lack of terms for time as an abstract concept and traditional cognitive linguistic views of time as a backdrop against which events can occur, a concept the authors call "Time as Such".

> "The term for 'day' in Amondawa, Ara, refers only to the daylight hours and also has the meaning 'sunlight'. There is no Amondawa term for the entire 24-hour diurnal cycle. Ara, 'day', contrasts with Iputunahim, 'night', which also means 'intense black'. There is a major subdivision of Ara, 'day', into two parts, Ko' ema 'morning', and Karoete 'noon/afternoon'. Thus, additionally to the binary day-night contrast, it is also possible to say that the 24-hour period is divided into three major parts, Ko' ema, Karoete and Iputunahim. Both day and night are further subdivided into intervals which are conceptualized and named on the basis of the daily round of activities." (Sinha et al., 2011)

Sinha et al. (2011) conclude that the Amondawa speaking people have a conceptualization of time governed by events. They define event-based time intervals as ones whose boundaries are constituted by the event itself.

The Lingodroids in our studies are a step toward robots that would have the appropriate architecture to learn Amondawa time concepts. A benefit of using Amondawa time concepts is that events may be decoupled from clock time. A task may need to happen in conjunction with a particular event each day, and referring to the task using clock time would be incorrect in general. For example, scheduling breakfast at sunrise every day cannot be expressed using clock time without continual recalculation.

One of the goals of the Lingodroid studies is to incorporate sufficient capabilities so that the robots could learn concepts from any natural language. The results of the current studies suggest that event-based time concepts are an aspect of natural language that can be easily learned through direct experiences. In our previous studies of the grounding of durations in journeys, we observed that space, time and change are inter-related aspects of navigation tasks and terms for all three types of concept can be grounded directly in experience (Schulz et al., 2011b). The current studies show how event-based experiences can be used for establishing shared attention for temporal concepts.

In analyzing what changes were required to the Lingodroid architecture to learn these new concepts, many of the issues, noted in previous studies (by our research group (Schulz et al., 2011b; Schulz et al., 2011) and others (Vogt, 2002; Steels, 2005; Roy, 2005)), were also important in these studies, including social interaction, the establishment of shared attention, appropriate representations, and practical tasks.

The social interactions by the Lingodroids are critical to how concepts are shared and meanings for words are grounded. Establishing shared attention to features of the environment (such as sunlight) was designed into the social interactions in these studies. An open question is how to extend the interactions to enable the robots themselves to indicate novel aspects of their experiences to attend to, and then to use these aspects in shared tasks. The issue is not just how to get a robot to attend to the level of sunlight, but also how to use the feature as a meeting time.

The studies suggest some characteristics for successful event-based temporal lexicons. Features such as sunlight levels alone would not have constituted a successful basis for the time-of-day concepts. The success also required the derivative of the light levels, whether the light was waxing as an indication of morning or waning as an indication of afternoon. Galton (2011) emphasized that aspects of time such as extent, linearity, and directness can be shared with space, but the transience of time cannot. In our previous studies of time mapping durations into journeys, we were able to focus on the shared experience of the journey, rather than the transient quality of time *per se* (Schulz et al., 2011b). In this chapter, transience is implicit in the changing light levels.

Another feature of events that are useful for grounding time and planning is predictability. Cyclical events may be well suited to the planning of cyclic tasks; however, any predictable event may be used to plan a scheduled task. Although rare or even impossible events may also be used in human language (e.g., "once in a blue moon," "when the seas run dry," "when pigs fly"), they are not used to plan events but rather as metaphors with other communicative intent.

In the current Lingodroid studies, the temporal lexicons have more words for more rapidly changing features. The robots developed three words for times of day, and only one for night. This difference is a result of "night" only having one value, total black, in the iRat environment. These results are reminiscent of the Amondawa-based subdivisions, which include two main words for day and only one for night (Sinha et al., 2011). While the current studies did not extend to motion, we conjecture that similar characteristics may also apply to motion words.

Another aspect of the robot studies is the importance of the fully embodied studies. The robots' behavior is subject to various sources of noise, some are statistical and can be estimated by Gaussian noise, while others are due to non-random aspects of the world. A challenge in the studies was the iRats' estimations of the time to reach their goals, and to navigate around the other robot. The primary robot behaviors used for constructing RatSLAM maps is wall following. However, in the real environments, this means that there is a high likelihood that the other robot, due to its own wall following behaviors, will be on the same path. In the *meet-at* task, a clear problem occurs when one robot reaches its intended destination, and thereby blocks the other robot from reaching its goal.

The embodiment of the iRats, in particular their forward-facing cameras, differed substantially from the pioneer robots used in previous Lingodroid spatial language studies. A potential problem was that the view-based matching in RatSLAM would be affected by the presence of the other iRat in the visual field

during conversations. This was not a problem for the current studies, as the iRats were set to use only the upper parts of their visual field as the most informative region for localization.

5.5.1 Limitations

The studies are a first step toward event-based temporal terms, and we recognize a range of simplifications which could be addressed in future work. Some of these limitations are due to the event-based time phenomenon itself while others arise due to the practical considerations of running fully embodied robot experiments.

Use of simulated light levels: The studies used a simulated vector instead of dynamically changing light levels as a pragmatic decision to enable studies to be run in the everyday working environment of a lab. Using a simulated vector also enabled the Lingodroids' standard mapping system, RatSLAM, to be used for the studies. Recent developments with the RatSLAM algorithm may enable this restriction to be relaxed in future studies (Glover et al., 2010).

Symbol grounding versus language: It should be noted that these studies concern the grounding of temporal concepts, rather than fully fledged languages. The emergence of linguistic structure is outside the scope of this work. For studies of grammar, syntax, and generational learning, see (Steels, 2000; Kirby, 2002; Cangelosi et al., 2010; Uno et al., 2011).

Duration estimation: Coordination requiring any form of duration estimation is challenging using event-based temporal terms. In these studies, to meet at a particular event-based time, the Lingodroids had to decide when to start moving toward the meeting location. This problem may sound simple in standard navigation, involving calculation of the distance to the goal, estimation of travel time required for that distance, and estimation of departure time by subtracting travel time from the desired meeting time. When using sunlight grounded time, the problem is more challenging. The Lingodroids can estimate distances, but using events alone, they cannot estimate travel times. An interesting conjecture from this project is that similar difficulties could be experienced by the Amondawa-speaking people. It would be interesting to know how they schedule important events that require estimations of durations such as travel times, and whether they have a separate set of terms for durations (Sinha et al., 2011).

Calculation of brightness derivative: The brightness derivatives were calculated directly from the sunlight equation. For real world experiences, it will be necessary for the robots to deal with noisy signals for light levels and averaging of light levels to create a gradient. One approach would be to estimate the gradient information over time, learning how fast features can change, and what time scales are useful for the robots in their daily interactions.

5.6 Conclusion and future work

For robots to learn the many and varied meanings of time, they need to be able to not only ground the meaning of time words in the clock, but also be able to do so with the features of the world. These studies are the first to use embodied robots to develop temporal terms grounded in both the features of the environment (sunlight) and in the transient quality of changing light levels (the sunlight derivative). This unique grounding allows for a dynamically changing representation of time that in turn allows the scheduling of tasks that require alignment with events rather than clock time.

These studies also demonstrate that scheduling meeting tasks using sunlight grounded time has a high success rate and is useful when describing a task that needs to co-occur with a particular event.

Transience is inherent in natural language concepts for time. What it will take for robots to understand concepts for transience and develop a grounded lexicon that is not just used implicitly, as in the current Lingodroid studies, but also to draw the attention of other robots to it and to make it the focus of their conversations, has yet to be determined.

Chapter 6 – Study III

Communication between Lingodroids with different cognitive capabilities

---

As described in Chapter 1, mobile robots come in many different types, and for robots with different cognitive architectures to communicate successfully requires them to compensate for these cognitive differences. One of the goals of this thesis is to outline sufficiencies for grounded communication across different cognitive architectures. While a previous study has looked at spatial lexicon learning using robots with different sensors (Jung and Zelinsky, 2000) (see Section 2.3.8), grounded communication has not been previously studied using robots with different cognition.

The robots used in the studies in this chapter – the iRat and the laserbot – had different spatial sensors and spatial cognition. The iRat used a forward facing camera with RatSLAM, while the laserbot used a laser range finder with Gmapping and AMCL (see Section 2.3.1). The different sensors were used with appropriate cognition in both cases.

The studies in this chapter were motivated by the future requirements of heterogeneous robots working within teams. Teams of mobile robots can perform tasks that require spatial exploration, searching or ground coverage in parallel. Teams of heterogeneous robots will be able to introduce different sensors and cognition to searching and exploration and there will need to be a way to communicate these subjective experiences. Using learned, grounded symbols is the first step towards heterogeneous robots that can learn to communicate what is necessary for collaborative tasks.

Two studies were designed to investigate grounding terms for toponyms across different sensors and cognition. In the first study, the L2 robots developed spatial lexicons using their different SLAM systems and spatial representations. In the second study, the L2 robots used their spatial lexicons to bootstrap additional lexicons for distance and direction terms.

The *where-are-we* conversation (from Schulz et al. (2011a), see 3.3) was used to learn terms for toponyms grounded in different SLAM representations. The generative conversations *how-far* and *what-direction*, (from Schulz et al. (2011b)), were used to develop distances and directions. These conversations were modified for the laserbot to allow toponyms to be grounded in grid squares, and distances and directions to use the modified toponyms.

The studies described in this chapter were the first to use the Australia maze as the L2 robots' environment (see Section 3.1). In the first study the L2 robots, embodied as an iRat and laserbot, were placed in

the Australia maze and followed the exploration algorithm of Section 3.1.1. Whenever the robots entered shared attention, they initiated a *where-are-we* conversation. Unlike the studies of Chapters 4 and 5, the L2 robots with their different cognitive architectures were not able to directly compare their spatial lexicons, although visual inspection indicated similarities.

In the second study, the L2 robots held a series of *how-far* and *what-direction* conversations to develop lexicons for distance and direction terms. The distance lexicons and direction lexicons were then compared across the agents. Sets of 100 conversations were held for each of *how-far* and *what-direction* and the coherences were averaged over 10 trials. The Lingodroids attained an average of 78% coherence for distances and 72% coherence for directions. These results indicate that robots with different cognitive architectures can use grounding transfer (see Section 2.3.1 and Schulz et al. (2012)) to allow them to create identically grounded distance and direction terms based on differently grounded spatial terms.

Several conclusions were drawn from these studies. Firstly that it is possible for robots using the L2 framework to learn coherent lexicons for space even when the underlying spatial sensors and cognition are different. The changes required to the L2 framework are the additions of features grounded in grid squares and a distance metric for those features (Euclidean distance). The L2 robots use some prior knowledge, such as the same generalization radius, and it is not yet clear if such knowledge is required.

There are some known limits to the allowable differences between robots that are still capable of producing coherent lexicons. Both agents need a common process for learning shared symbols and both agents need referents that are *reliable* within the environment (i.e. repeatable for the same environment conditions).

Finally, these studies contest the assumption that shared understanding must be grounded in shared biology. For spatial lexicons, the L2 framework demonstrates that it is possible to bridge sensory and cognitive differences.

The following sections have been reproduced from:

- Heath, S., Ball, D., Schulz, R. and Wiles, J. (2013). Communication between Lingodroids with different cognitive capabilities. In *Proceedings of the International Conference on Robotics and Automation*, pages 490-495.

The sections have been taken from the final submitted manuscript and reformatted to fit within this thesis. Note: within this publication, the L2 framework is referred to as just Lingodroids.

Abstract – Previous studies have shown how Lingodroids, language learning mobile robots, learn terms for space and time, connecting their personal maps of the world to a publicly shared language. One caveat of previous studies was that the robots shared the same cognitive architecture, identical in all respects from sensors to mapping systems. In this chapter we investigate the question of how terms for space can be developed between robots that have fundamentally different sensors and spatial representations. In the real world, communication needs to occur between agents that have different embodiment and cognitive capabilities, including different sensors, different representations of the world, and different species (including humans). The novel aspects of these studies is that one robot uses a forward facing camera to estimate appearance and uses a biologically inspired continuous attractor network to generate a topological map; the other robot uses a laser scanner to estimate range and uses a probabilistic filter approach to generate an occupancy grid. The robots hold conversations in different locations to establish a shared language. Despite their different ways of sensing and mapping the world, the robots are able to create coherent lexicons for the space around them.

6.1 Introduction

A key challenge for performing useful tasks with a team of heterogeneous robots and humans is the ability to communicate effectively between agents with different embodiment and cognitive capabilities. The embodiment of an agent consists of its sensors, actuators and physical body, while cognitive capabilities include learning and language abilities. Lingodroids – language learning robots – have been used to model cognitive processes ranging from knowledge representation and planning to language development, symbol grounding and even imagination. A key aspect of lexicon learning is the connection of a word to its meaning, called "symbol grounding". The Lingodroids' advanced navigation skills have been particularly useful in learning lexicons that name spatial aspects of their environments. Practical symbol grounding studies to date have rarely examined populations of heterogeneous robots. Robots with different sensors but identical cognitive representations were studied in Jung and Zelinsky (2000). It is still an open question what level of communication can be achieved between agents with different cognitive capabilities.

Previous work in the Lingodroids project has shown that real robots can learn a language for human-like concepts of space (locations and spatial relationships (Schulz et al., 2011a)), and that this framework can be extended to temporal concepts of durations (Heath et al., 2012a). In previous Lingodroid studies, the robots constructed their maps independently, and so have had unique cognitive maps of the world, but within each study the robots were functionally identical, using the same type of physical robot and the same underlying mapping algorithms and behaviors.

In this chapter, we investigate Lingodroids with different embodiment and cognitive capabilities. Studies were performed with pairs of real robots. The robots were based on a rat-sized robot called the intelligent rat animat technology, or iRat, developed at the University of Queensland. While the iRats had the same physical size, actuators, and language systems, they differed in three key aspects:

1. **Sensors –** One robot used the iRat's single standard forward facing camera, which provided color images that are converted into appearances ('camera iRat'). The other robot instead used a 240 degree laser scanner that provided metric range information ('laser iRat').

2. **Algorithmic approach –** The camera iRat used the biologically-inspired RatSLAM system (Milford and Wyeth, 2010) which uses a continuous attractor network to filter appearances and self motion. The laser iRat used a probabilistic filter approach to localize where particles represent possible poses in the map (Fox et al., 1999).

3. **Spatial representations –** The camera iRat constructed and relaxed a semi-metric topological map, called the experience map. The laser iRat created occupancy grid maps offline from laser scan information, using particles to represent possible maps (Grisetti et al., 2007).

When agents have identical embodiment and cognitive capabilities, it is theoretically possible to transfer knowledge directly from one agent to the next. When such capacities differ, a direct transfer is no longer an option. Symbol grounding must be achieved through private grounding using different algorithms for each robot type, and through social grounding that is appropriate for all of the robot types.

In this chapter we show that Lingodroids with different sensor types and map representations can develop coherent symbols for places, distances, and directions. Lingodroids is a good candidate for facilitating communication between teams of heterogeneous robots and humans, due to the human-like concepts learned in previous studies and its coupling with state-of-the-art SLAM systems (Schulz et al., 2011a). The major contribution of this chapter is the demonstration of spatial language learning on real robots with different embodiment and cognitive capabilities.

The chapter presents a brief review of related work before providing details about the robot platform and algorithms for building maps and grounding language. We describe the experimental setup and present results that show the coherence of the spatial lexicons. The discussion focuses on the potential extensions of the methodology.

6.2 Literature review

Heterogeneous robot teams are a growing research area, involving robots with a variety of abilities interacting, cooperating, coordinating, and communicating. Examples of tasks for teams of robots include

environment mapping (Simmons et al., 2000), cooperative localization (Parker et al., 2004), search and rescue (Murphy et al., 2000), and decentralized environment modeling (Gil Jones et al., 2006). A key challenge for robots that are part of heterogeneous teams of robots and humans is how to communicate about information in their respective knowledge bases, formed through their individual interactions with the world. The shared language used for communication must be grounded in each robot's own representations, thus addressing the challenge of the symbol grounding problem referred to in the introduction (Harnad, 1990). To effectively communicate with each other, the robots need to link individual experiences with symbols via private or physical grounding (Brooks, 1990), and develop a standard usage of shared terms via social grounding (Cangelosi, 2006). One solution to the symbol grounding problem involves robots learning categories embedded in the robot's sensorimotor interactions by playing language games (Steels, 2001). Many variations on language games have been developed for different tasks and environments (Vogt, 2002; Kirby and Hurford, 2002; Cangelosi, 2006) including spatial locations and relations (Steels, 1995; Jung and Zelinsky, 2000; Cangelosi et al., 2005; Schulz et al., 2011). Cognitive capabilities for mobile robots may differ in the mapping systems used. Simultaneous Localization and Mapping (SLAM) systems can vary in the sensors and the mapping algorithms used. Two distinct approaches to solving the problem of SLAM are probabilistic approaches (Bailey et al., 2006; Grisetti et al., 2007) and biologically inspired approaches (Barrera and Weitzenfeld, 2008; Milford and Wyeth, 2010).

6.3 Method

This section describes the robots' mapping systems. Note that the robots use the same Lingodroid communication system; however, the connections differ from the lexicons to the maps.

6.3.1 Robot platforms and environment

The same base robot platform, the iRat (Ball et al., 2010), is used for both agents. The iRat is the same size and mass as a large rodent, with an onboard 1GHz computer and wireless 802.11g/n. The robot moves about its environment using a differential drive system. Sharp infrared sensors orientated at -45, 0 and 45 degrees provide range information. The robot runs the Robot Operating System (ROS) (Quigley et al., 2009) on Ubuntu and communicates to clients over wireless at 20Hz to send and receive desired and actual velocity commands.

For the studies in this chapter the iRat uses its infrared range sensors for avoiding obstacles such as the walls and other iRats. The iRat attempts to wall follow down the center of a corridor. When it arrives at an intersection it randomly chooses a direction for exploration. The two iRats have different primary sensors (see Figure 6.1). The camera iRat is unmodified from the standard build and uses a forward facing wide-screen camera with a horizontal field of view of 110 degrees. A ROS node publishes compressed 416x240 pixel color JPEG images from the robot's forward facing camera.

The second iRat, the laser iRat, has had the camera replaced with a Hokuyo URG laser sensor that is mounted upside-down so that it can sense the range to the walls of the environment. The laser scanner has a scan angle of 240 degrees with an angular resolution of 0.36 degrees and a detection range of between 20mm and 1000mm. A standard ROS node driver is used to publish the laser scans.

Figure 6.1: Lingodroid iRats in conversation. On the right is a standard 'camera iRat' which has a forward facing camera inside the dot of the 'i'. On the left is the modified 'laser iRat' which has a laser scanner mounted upside down. The standard iRat's cover has been removed to fit the laser scanner. The two iRats are shown in the environment used for the studies in the chapter (also see Figure 6.2).

These studies were performed within an environment made specifically for the iRats, modeled on a map of Australia. An overhead view of the set, which measures 3.2 x 2.4 meters is shown in Figure 6.2. Prominent Australian features such as Uluru (the red rock in the center) and the Sydney opera house (bottom right) can be seen.

6.3.2 SLAM systems

The robots' SLAM systems are completely different, one based on a biologically inspired topological approach, the other based on a probabilistic metric approach. Previous Lingodroid studies have only used the biological system.

The biological system is called RatSLAM and is inspired by the rodent hippocampus (Milford and Wyeth, 2010). It has three parts: local views, pose network, and experience map relaxation. The local view part uses the camera images to match locations based on appearance similarity. The pose network is an energy based continuous attractor network which filters local view appearances and self motion estimates. The experience map builds and relaxes a semi-metric topological map. Although an initial map is created at the beginning of the study, the RatSLAM map is continuously corrected during usage.

Figure 6.2: The environment used for the studies in the chapter, modeled after a map of Australia. The two iRats are shown interacting in the center right of the image.

The probabilistic system uses Gmapping (Grisetti et al., 2007) for map construction and Adaptive Monte-Carlo Localization (AMCL) (Fox et al., 1999) for robot localization within the map. Gmapping uses a particle filter to build occupancy grids from metric range and self motion information. Each particle carries an individual map of the environment, and the filter attempts to reduce the number of particles. The occupancy grid is constructed at the beginning of the study. The robot is then localized during studies using AMCL, a system which maintains a probability distribution using particles over possible robot poses within a static occupancy grid. The particle distribution is spread and moved based on robot motion and re-sampled based on laser range data. AMCL requires a relatively accurate estimate of the robot's motion, which is not met by the iRat's coarse wheel odometry. Instead, local laser scan matching (Censei, 2008) is used to provide the motion estimate. Note that RatSLAM tolerates the inexact wheel odometry due to the use of a topological map.

To counter the uncertainty in localization using AMCL with the iRats, the laser iRat did not consider localization accurate if AMCL's global particle covariance was too high (determined by summing the covariance matrix elements and setting a threshold of 0.1 for these studies).

The important differences between the two mapping systems from a grounding perspective are the output representations. The output representation of RatSLAM is a topological map, expressed as a graph,

whereas the output representation of Gmapping is an occupancy grid, expressed as a 2D array (see Figure 6.3). The output of Gmapping remains static after the initial map-creation phase; however, the output of RatSLAM continues to change and correct, as discussed further in the next section. Gmapping provides additional information about the location of obstacles compared with the map provided by RatSLAM. The occupancy grid produced by Gmapping will be 'metric' (distances and directions will be very accurate), while RatSLAM provides only a semi-metric map (distances and directions are only partially accurate).

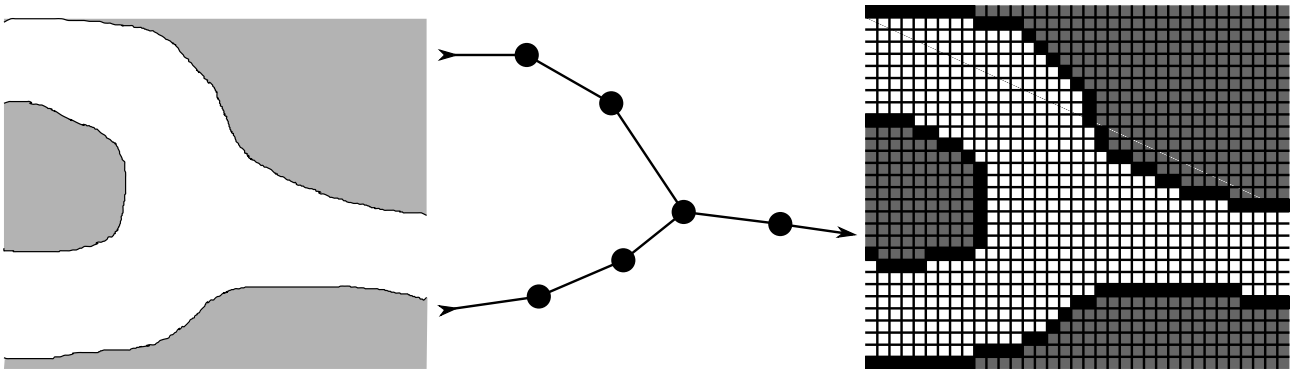


Figure 6.3: Topological representation vs occupancy grid representation - a) the overhead view of a robot's environment, b) a topological representation c) a occupancy grid representation.

6.3.3 Language platform: Lingodroids

Lingodroids develop lexicons using pairs of robots to evolve a shared language over a series of conversations, which are short social interactions (Schulz et al., 2011). The most basic conversations consist of a single question and response. The robots converse to learn words for locations called toponyms, which literally means 'place names'.

The robots use *where-are-we* conversations to create names for shared locations. Over many such conversations, the robots agree on a set of toponyms referring to different locations in the environment. This toponym lexicon can then be used to bootstrap generative conversations, such as *how-far* and *what-direction* – conversations that allow the Lingodroids to form a set of lengths and directions corresponding to the distances and angles between toponyms. This bootstrapping involves indirectly grounding the elements, a process known as 'grounding transfer' (Cangelosi et al., 2000).

Words are associated with component parts (called 'concept elements') of the toponyms, distances, and directions. For the toponyms, different location-related concept elements were used for each SLAM system: for the camera iRat, toponyms were associated with RatSLAM experiences; for the laser iRat, toponyms were associated with grid square locations in an occupancy grid. RatSLAM experiences record positions in meters in abstract space while the grid squares are recorded in pixels. A key difference between the two types of concept elements (experiences vs pixels) is that RatSLAM experiences can move, as the map is continually corrected during usage. Having the concept elements move allows words to change meanings to correct associations formed when the robot was incorrectly localized. The occupancy grid created using Gmapping does not change during use, therefore words associated with that map never

change meaning. For both systems, distance and direction concept elements are created as needed when referred to during a conversation. Distances are calculated from the estimated metric distance between locations in each robot's map. Directions are calculated from the angle between three locations in each robot's map.

Associations between words and concept elements are stored in distributed lexicon tables. Distributed lexicon tables maintain a set of all of the words used by either agent during conversations, a set of all the concept elements and a set of edges between words and concept elements that have been used together, called associations. This data structure allows storage of many-to-many relationships between words and concept elements. The strength of an association is the number of times that its $<$ word, concept element $>$ pair has been used together in conversation. The resulting structure allows a single trial to define a word, with additional trials refining the use of a word. When choosing a word to describe a concept element, the speaker finds the word-concept element pair with the highest confidence. The confidence value, h_{ij}, for a word, j, and a concept element, i, is calculated by:

$$h_{ij} = \frac{\sum_{k=1}^{X} a_{kj} \left(D - d_{ki}\right) / D}{\sum_{m=1}^{N} a_{mj}}, \tag{6.1}$$

where X is the number of concept elements within a neighborhood of size D of the current concept element, i; a_{ij} is the number of times that the concept element, i, and the word, j, have been used together; d_{ki} is the distance between concept element k and i; and N is the total number of concept elements. The distance between two concept elements is defined as the Euclidean distance between the coordinates of the elements.

The neighborhood size determines the coverage of the toponyms formed in each map: For toponyms and distances it was set to 0.3m for the camera iRat and 100px for the laser iRat, and for directions for both systems it was set to 60°.

Words are invented with probability, p, using the confidence value and a word invention temperature, as follows

$$p = k \exp\left(\frac{-h_{ij}}{\left(1 - h_{ij}\right) T}\right), \tag{6.2}$$

where $k = 1$, h_{ij} is the confidence value of the concept element-word combination, and T is the temperature, the word invention rate. A higher temperature causes words to be invented with higher frequency even when a valid generalization is available. T was set to 0.1 in this study for all concept types.

Lingodroids' conversations are started by the robots whenever they have shared attention. Shared attention is established by an overhead camera, which detects when the two robots are within 50 pixels of each other, or 0.25m apart in the environment, and notifies the two agents simultaneously with a boolean value.

Conversations using grounding transfer, such as *how-far* and *what-direction* do not depend on the robots' physical locations and so may be performed offline.

6.3.4 Quality measures

Previous Lingodroid studies use coherence between lexicons as a quality measure. Coherence is calculated by rendering a lexicon onto a fixed resolution grid and then determining the number of matching grid squares as a percentage. However, in these studies it is not possible to calculate coherence of the toponymic lexicons directly, as the two mapping systems are not commensurate, in that features stored by one SLAM system have no representation in the other system.

The coherence of the two toponymic lexicons were instead established by calculating the coherence of distance and direction lexicons that were constructed from them. Coherence was calculated for distance lexicons by choosing words to describe distances at 100 points from 0m up to the maximum distance in each robot's lexicon. The percent of matching words was then calculated between the two robots. For the direction lexicons, the word chosen was determined for every 2.5° from 0° to 360°. If the toponym lexicon is coherent and well grounded by each robot, then the *how-far* and *what-direction* conversations are likely to be coherent too.

6.4 Experimental setup

1. Map building - Both iRats independently explored the set and built their maps prior to starting the conversations. The camera iRat created a topological map using RatSLAM and the laser iRat created an occupancy grid map using Gmapping. The maps were saved at the end of this phase.

2. Learning toponyms - The two iRats were then placed back into the set together, the camera iRat running RatSLAM on the previously created topological map and the laser iRat running AMCL on the previously created occupancy grid. The iRats explored the set for two hours, localizing and holding *where-are-we* conversations when they moved within shared attention range. The conversations were used to create independent toponym lexicons.

3. Learning distances and directions - Distance and direction conversations were performed offline using the created maps and toponym lexicons from the previous phase to allow the creation of separate lexicons for distances and directions. 10 trials were done for this phase starting with the same initial maps and toponym lexicons from phase two. 100 *how-far* and 100 *what-direction* conversations were held for each trial. Average coherences were calculated across the 10 trials.

6.5 Results

After their initial explorations, each robot had individually mapped the area. After the *where-are-we* conversations, they had together constructed a shared toponymic lexicon (see Figure 6.4). A total of 10 toponyms were created, with most (8/10) toponyms covering a contiguous region of the environment, and two toponyms (lenu and kumu) covering two local regions separated by an intervening toponym.

Visual inspection indicates that the locations of toponyms in each map are similar; however, the maps from the two robots cannot be directly equated, since one is an occupancy grid and the other is a topological

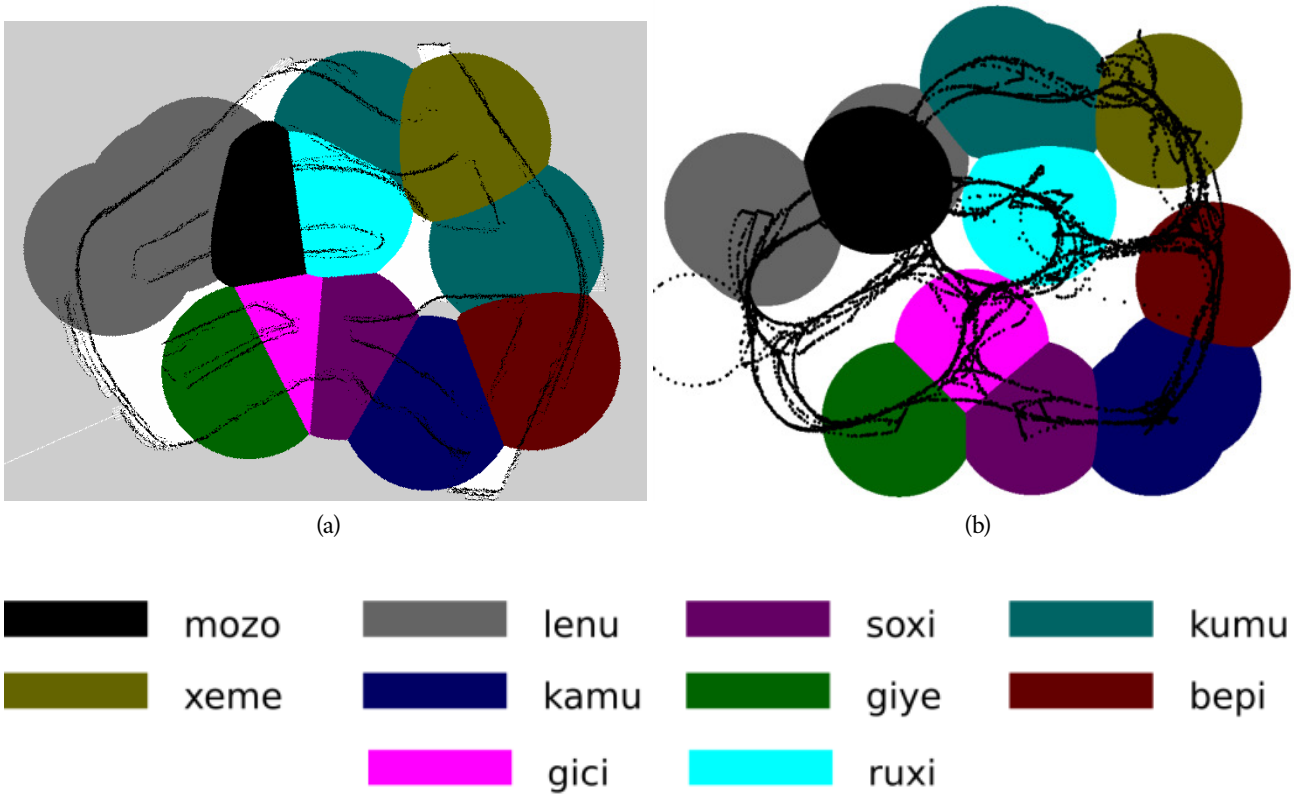


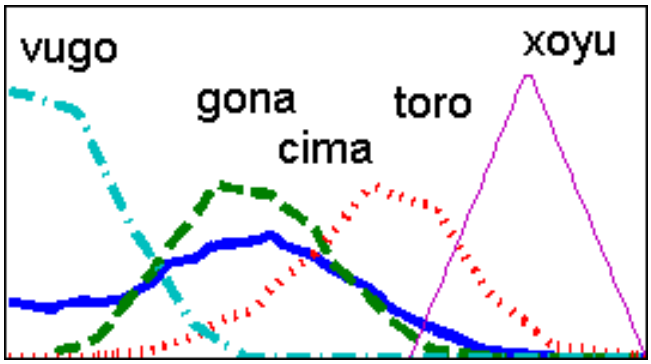
Figure 6.4: Maps and toponymic lexicons developed for both a) the laserbot, and b) the iRat. The maps of both robots are recognizably the environment with a high degree of similarity between the locations for all of the toponyms.

Table 6.1: Coherence of distance and direction lexicons

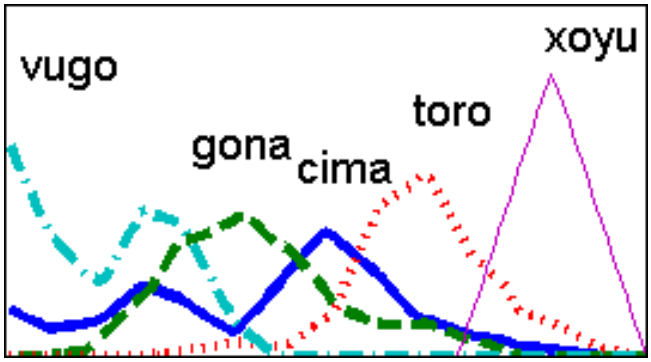
Measure	Distance (10 trials) $\mu(\sigma)$	Direction (10 trials) $\mu(\sigma)$
Coherence	$0.78(0.14)$	$0.72(0.14)$
Maximum coherence	0.95	0.94
Number of words	$4.2(1.1)$	$4.5(0.9)$
Number of concept elements	$9.7(0.88)$	$31.9(2.4)$

graph. What can be analyzed is the robots' functional use of the lexicon, by equating robot journeys on the maps. For example, to describe a journey starting in the north and following a clockwise journey around the outer perimeter of the set, each robot will pass through an almost identical sequence of terms. The edit distance between these two journeys is two (omission of kumu and addition of a second lenu in the camera iRat).

Following the 100 *how-far* and *what-direction* conversations, the robots had developed coherent distance and direction lexicons, with an average of 4.2 distance words and 4.5 direction words (averaged over 10 runs, see Table 6.1, Figure 6.5, and Figure 6.6). The average coherence of the distance lexicons was 0.78 and the direction lexicons was 0.72.

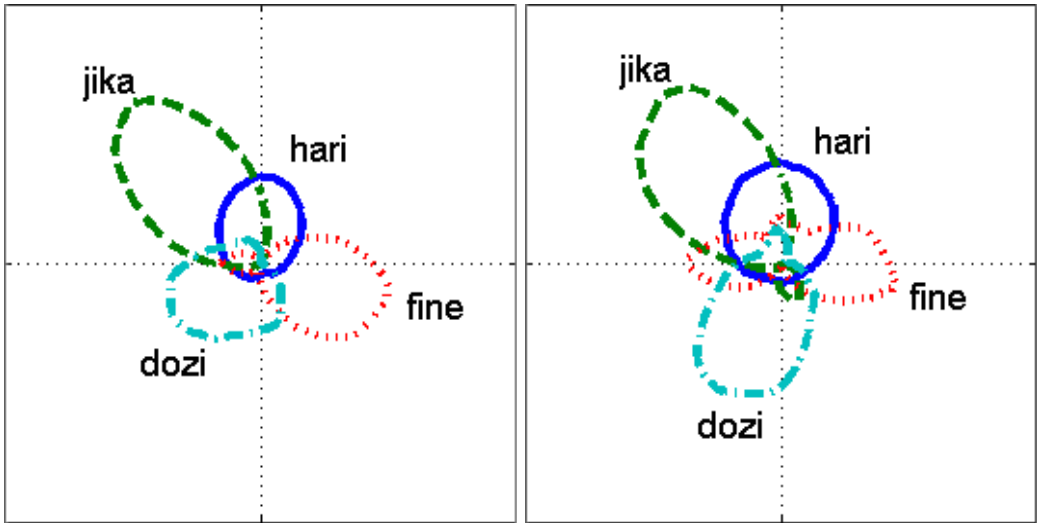


(a)



(b)

Figure 6.5: The most coherent distance lexicons (coherence of 0.95) for a) the laserbot, and b) the iRat.



(a) (b)

Figure 6.6: The most coherent direction lexicons (coherence of 0.94) for a) the laserbot, and b) the iRat.

6.6 Discussion and conclusions

These results show how communication can be achieved between agents with different sensors and mapping systems through shared externally grounded symbols. Experiences for each agent give rise to subjective characteristics that cannot be shared between them, but that does not preclude the evolution of a language for describing the world around them.

Toponyms learned by the Lingodroids are grounded in the very different representations of space formed from their characteristic sensors (laser vs vision). Features of these systems cannot be shared by direct transfer, nor can they interpret each other's maps. Instead, the two Lingodroids are able to ground their respective representations in shared experience. Each robot uses the shared experience to determine its own appropriate features and map.

It is likely that the more similar two agents are, the closer their subjective experiences will be (Nagel, 1974). However, even in previous Lingodroid studies with almost identical architectures (Schulz et al., 2011), direct transfer would still fail due to the subtle differences between the robots' sensors.

There are limits to the differences between agents using the Lingodroids methodology. In common with other robot language studies, agents must share a common process for learning a shared symbol, including hearer and speaker roles and sharing attention (Jung and Zelinsky, 2000) and each agent's referents must be reliable (Jung and Zelinsky, 2000; Steels, 2001; Vogt, 2002).

Granularity is an important consideration when dealing with continuous concepts, such as space or time (Varzi, 2007). An underlying assumption of the Lingodroids studies is that the reference of a location term extends to a radius around specific points where the term has been used (Schulz et al., 2011a). This simple version of location proves sufficient for bootstrapping distance and direction relations, and it is expected that with the same representations, the Lingodroids could be extended with minor modifications to learn spatial propositions typically studied in grid worlds (Steels, 1995; Cangelosi et al., 2005). However, more complex concepts of location can refer to the space occupied by an object or some superset or subset of that space (Varzi, 2007). A constraint on using the Lingodroids methodology is that when attending a location the relevant spatial referent must be shared by both agents.

One of the fundamental assumptions in human language used to be that shared understanding must be grounded in shared biology. The Lingodroid studies reported here complement human-robot studies in demonstrating for the most fundamental of all lexicons – describing practical terms for space – it is possible to bridge communication barriers across agents with different cognitive capabilities.

Chapter 7 – Study IV

Lingodroids: Cross-situational learning for episodic elements

---

Resolving uncertainty between symbols and meanings is an important part of language learning, and critical for bootstrapping language. As reviewed in Chapter 2, previous studies into resolving uncertainty had limitations for grounded learning, as the models were often based on ungrounded symbols, or symbols grounded in perceptual referents. The studies described in this chapter introduced cross-situational learning to the L2 framework, allowing the agents to resolve uncertainty between symbols and meanings over a number of "situations" (see Section 7.3.6). An extended framework was developed for this study that incorporated cross-situational learning, the grounded spatial learning from previous Lingodroid studies (Schulz et al., 2011a), the grounded temporal learning described in Chapter 4, and the grounded learning across different cognitive architectures described in Chapter 6. The studies described in this Chapter addressed all the thesis goals of Chapter 1: i) grounding symbols in spatial and temporal cognition, ii) grounding symbols across different cognitive architectures, and iii) resolution of uncertainty about links between symbols and meanings.

A single comprehensive study was designed to investigate how the addition of cross-situational learning affected the L2 framework's lexicon learning times and quality. In this study the L2 robots i) developed a hybrid lexicon that contained both spatial and temporal terms, ii) investigated the immediate usability of the lexicons – the ability to use the lexicons during learning, and iii) tested the lexicons on theoretical tasks.

A new conversation was designed, *where-in-space-time-are-we*, to share and ground toponyms and durations. The new conversation admitted multiple responses of both spatial and temporal symbols to be linked to both the current toponym and duration. These symbols could then be resolved when required by using KL-divergence to decide which symbol provided most information about a feature.

Two conditions were established in this study, the first – the Innate Condition – was based on the L2 that learned as per previous studies in Chapters 4 and 6. The robots in this condition used the conversations *where-are-we* and *when-did-we-last-meet* to develop one lexicon for space and one for time. The novelty in this study was in the second condition – the XS Condition, which extended the L2 framework with cross-situational learning. The robots in this condition used the new conversation *where-in-space-time-are-we* to develop a single hybrid lexicon for both space and time.

This study was run entirely in simulation. The L2 robots in this study were embodied as simulated versions of the iRat and laserbot used in Study III (Chapter 6). The simulated environment was based on that of the Australia maze (see Section 3.1). The robots first moved around the environment autonomously, following the same exploration algorithm from Section 3.1.1. When the robots had shared attention (given by simulated proximity), they initiated a *where-in-space-time-are-we* conversation (for the XS Condition) or *where-are-we* and *when-did-we-last-meet* conversations (for the Innate Condition) to develop their lexicons. For the XS Condition, the change in the KL-divergence for each symbol was monitored over time. A symbol's uncertainty at a point in time was judged by its informativeness ratio between toponyms and durations. The results indicate that typically, words start with very high informativeness for both space and time but a ratio of 1:1. However, over the course of the study, the informativeness decreased in both space and time, but the ratio moved towards just one of space or time.

The lexicons developed by the L2 robots were tested on a theoretical game based on the *meet-at* game from previous studies (see Section 3.3). The results indicated that the average difference in calculated arrival times and locations would be on average 4.5 seconds and 0.17 meters for the XS Condition and 11.9 seconds and 0.12 meters for the Innate Condition (additional results are included in Appendix B for completeness).

A key challenge of this study was the integration of all the thesis goals. Grounding in cognition, grounding across different cognitive architectures, and dealing with uncertainty between symbol and meaning are all dependent on each other. The generalization and representations of space directly affected cross-situational learning, while the decision on using a term as a toponym or duration in turn affected the generalization of that term and other competing terms. This study took into account these interdependencies when constructing an integrated framework.

Several conclusions were drawn from this study. The L2 robots were able to learn coherent lexicons after the addition of cross-situational learning, although immediate usability decreased and time taken for learning increased. Changes were required to aspects of the L2 framework – additions included a new conversation, hybrid lexicons, extra terms for resolving uncertainty and restricting terms from use when they were not well understood. However, core Lingodroids and L2 features also remained unchanged, such as the storing of exemplars in lexicons, the comprehension, production and generalization within a single dimension, and word invention.

The lexicons developed in this study provide an interesting perspective on a word and its meaning. Although it has been suggested before that different agents in a population have different cognitive representations of the same word (Steels, 2015), the Lingodroids in this study demonstrate different cognitive representations of words that will converge to the same meaning, but will not converge to the same representation.

The following sections have been reproduced from:

- Heath, S., Ball, D. and Wiles, J. (In Press, submitted 2015). Lingodroids: Cross-situational learning for episodic elements. *IEEE Transactions on Autonomous Mental Development*

The sections have been taken from the final submitted manuscript and reformatted to fit within this thesis. Note: within this publication, the L2 framework is referred to as just Lingodroids.

Abstract – For robots to effectively bootstrap the acquisition of language, they must handle referential uncertainty – the problem of deciding what meaning to ascribe to a given word. Typically when socially grounding terms for space and time, the underlying sensor or representation was specified within the grammar of a conversation, which constrained language learning to words for innate features. In this chapter we demonstrate that cross-situational learning resolves the issues of referential uncertainty for bootstrapping a language for episodic space and time; therefore removing the need to specify the underlying sensors or representations *a priori*. The requirements for robots to be able to link words to their designated meanings are presented and analyzed within the Lingodroids – language learning robots – framework. We present a study that compares pre-determined associations given *a priori* against unconstrained learning using cross-situational learning. This study investigates the long-term coherence, immediate usability and learning time for each condition. Results demonstrate that for unconstrained learning, the long-term coherence is unaffected, though at the cost of increased learning time and hence decreased immediate usability.

7.1 Introduction

Space and time are fundamental aspects of human languages, used for communicating real-time experiences and episodic memories (Boroditsky, 2001; Levinson, 2003). For robots to communicate spatial and temporal information, they will need the ability to learn referents for spatial and temporal words.

To date, robots have independently learned words for space and time, resolving word meanings using referents determined by the dimensions encoded in the grammar of a conversation (Schulz et al., 2011a) or through previously specified meanings (Jung and Zelinsky, 2000). These solutions limit the utility of robot language learners to what is encoded *a priori*.

To enable robots to learn to talk about episodic events without *a priori* encoded dimensions, it is useful to develop learning algorithms that resolve the dimension (space, time, or both) as part of the determination of a word's specific meaning. In ambiguous contexts the major approach to resolving dimensions has involved extracting regularities across multiple examples of each word, a process called cross-situational learning (XSL) (Siskind, 1996).

Practical XSL studies can resolve the dimensions for perceptual referents (Roy, 2002a) and within ungrounded simulations (Siskind, 1996; Smith et al., 2006; Blythe et al., 2010); however, perceptual referents do not include the more abstract concepts of space or time, or their cognitive representations such as mental maps. These abstract concepts form the foundations of a robot's cognition and are the key building blocks of episodic memories. Symbol grounding studies have explored grounding in cognitive processes for learning terms for space, time and actions, but within these studies, resolving dimensions has largely been ignored (Jung and Zelinsky, 2000; Marocco et al., 2010; Schulz et al., 2011a).

XSL has never been tested on the mental maps and temporal cognition that are required for communicating episodic space and time. Important factors in the use of XSL for resolving spatial and temporal dimensions are the effects on language learning performance, the changes to the quality of the learned language and the time taken to learn the language.

This chapter addresses the resolution of dimensions for episodic space and time with the goal of reducing reliance on predetermined dimensions. Limiting predetermined dimensions is particularly important for heterogeneous robots, where different sensors and cognition requires assumptions about correlations between innate representations of sensory data.

Our specific aim is to examine language learning using XSL with episodic elements. We introduce a new extended Lingodroids framework and compare it to the existing Lingodroids framework on language usability and learning time for building lexicons using a medium-fidelity simulation based on robots from our previous studies.

7.1.1 Symbol grounding and cross-situational learning

For robots to *understand* language, the symbols that make up the language must be *grounded*. *Symbol grounding* is the association of a symbol with a meaning. A symbol can be any perceptible entity, although typical media are sound (words and syllables) (Steels, 1999) and light (pictures and text) (Galantucci, 2005). Symbols in communication can act as aliases for other symbols or as proxies for objects in an agent's environment. The *symbol grounding problem* refers to the infinite recursion caused by grounding symbols in other symbols (Harnad, 1990). Symbols must instead be eventually grounded in the sensors of an agent; however, this grounding may be direct (e.g. symbol $\rightarrow$ sensors), indirect (e.g. symbol $\rightarrow$ symbol $\rightarrow$ sensors) or through abstraction (e.g. symbol $\rightarrow$ abstraction $\rightarrow$ sensors), as is the case for space and time.

The semiotic relationship between a symbol and its meaning involves two steps: i) private grounding, where an agent creates an internal representation for a perceptual referent, and ii) social grounding, where two or more agents share a symbol that refers to their internal representations (see Figure 7.1; Cangelosi (2006) and Schulz et al. (2011a), adapted from Ogden and Richards (1923)).

Resolving the dimensions of referents is a key problem associated with social grounding, a problem typically called *referential uncertainty*. Referential uncertainty refers to the inability of an agent to unambiguously associate a word with its meaning due to multiple candidate associations in the agent's context. For example, if a native pointed at a rabbit and said "gavagai", someone listening would not know if the native was referring to the rabbit, parts of the rabbit or even something unrelated to the rabbit (Quine, 1960). An infinite number of meanings can be associated with any given context.

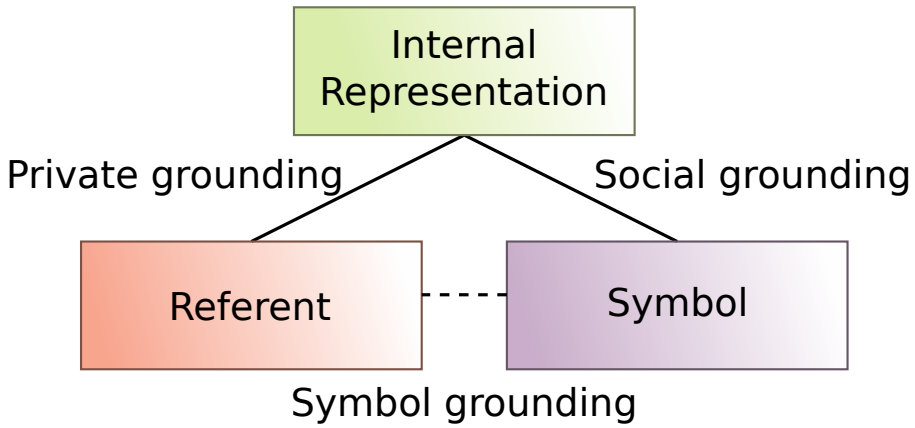


Figure 7.1: Grounding and the semiotic triangle. A symbol is linked to a referent through social grounding (agents agreeing on a symbol) and private grounding (forming an internal representation). Figure adapted from Ogden and Richards (1923).

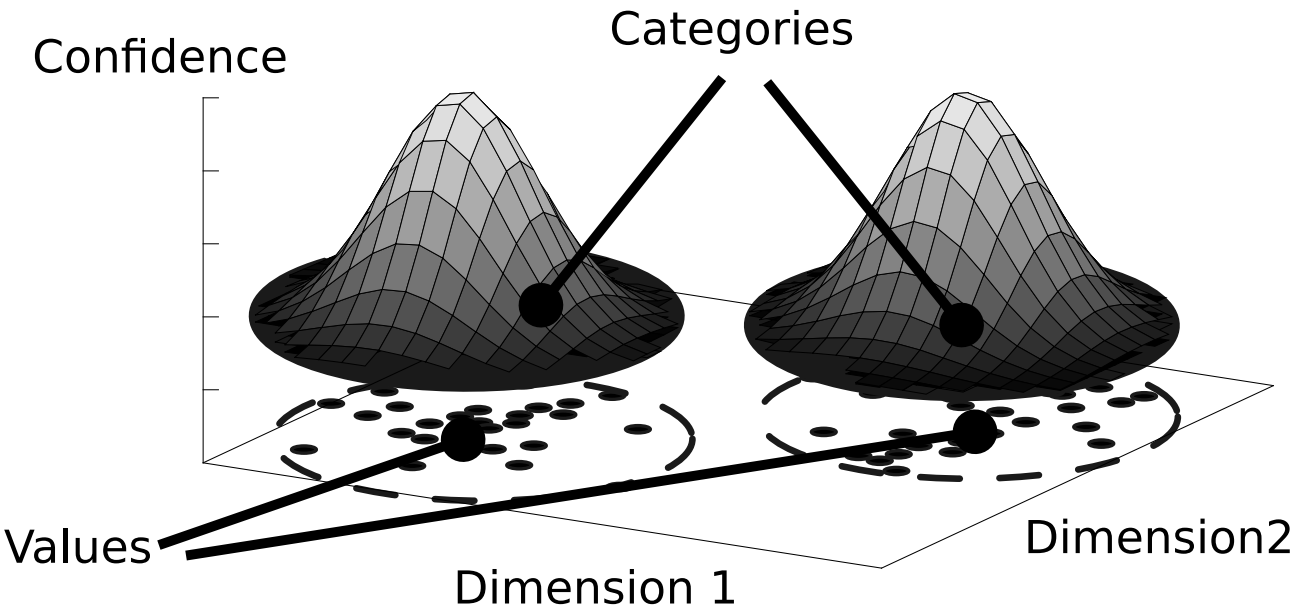
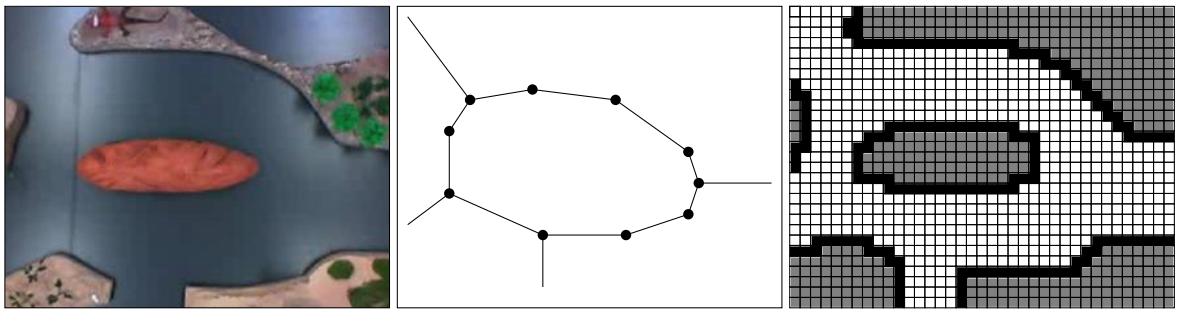


Figure 7.2: A common framework for XSL with continuous features. Values (or concept elements in Lingodroids) are expressed as a point in N-dimensional space and linked to direct evidence experienced by a robot.

Referential uncertainty is closely linked to the ability of agents to share attention. Mechanisms, such as restricting a context to be unambiguous (i.e. words for which shared attention is identical and link to a single referent) (Jung and Zelinsky, 2000), using additional language to specify attention (Schulz et al., 2011a) and XSL are different methods of controlling referential uncertainty that have been used in previous studies. The advantage of XSL is that less *a priori* knowledge is required compared to other methods. Learning only unambiguous words requires reducing the contexts through prior knowledge, and additional language requires that both robots already have the symbols to refer to objects in their contexts. Previous XSL studies for continuous features are set within a framework of examples, categories and dimensions (see Figure 7.2). To our knowledge this is the first study to use XSL to discriminate between referents that are cognitive representations of space and time.



(a) (b) (c)

Figure 7.3: Topological representation vs occupancy grid representation. a) the overhead view of a robot's environment, b) a topological representation, and c) an occupancy grid representation.

7.1.2 Different perspectives of space

Although space and time have strict physical definitions (that are mathematically related) (Feynman, 1948), space and time are expressed differently in natural language (Clark, 1973; Engberg-Pedersen, 1999; Varzi, 2007; Kuipers, 2008). To develop and communicate using these different expressions, robots require appropriate spatial and temporal cognition.

In this study, two state-of-the-art Simultaneous Localization and Mapping (SLAM) systems are used to provide two different spatial representations: topological and occupancy grid (see Montemerlo and Thrun (2007b) for an introduction to SLAM). Topological representations store maps as a graph where nodes are points of interest and edges are the agent's trajectories, while occupancy grids represent obstacles, free space and unknown areas as different colors in a bitmap (see Figure 7.3). The bio-inspired mapping system RatSLAM is used to provide a topological map from an agent's camera images and odometry (Milford and Wyeth, 2010). The particle filter-based Gmapping is used to provide an occupancy grid representation from an agent's laser range scans and odometry (Grisetti et al., 2007).

RatSLAM and Gmapping are typically used in different ways. The map created by RatSLAM continues to evolve during the course of usage; however the Gmapping map is not changed once created. Instead, Adaptive Monte-Carlo Localization (AMCL) (Fox et al., 1999) is used to localize using the Gmapping map.

The RatSLAM and Gmapping / AMCL SLAM systems provide cognitive maps for the robots that allow them to ground spatial language in cognition, instead of directly in perception. SLAM systems can encode the spatial elements of episodes, allowing robots to refer to specific places and maintain relationships between places.

7.1.3 Temporal cognition

Temporal cognition is increasingly recognized as an important attribute for artificial intelligence (Maniadakis and Trahanias, 2011). Several computational models of temporal cognition in the human brain have been developed (Taatgen et al., 2007; Choe et al., 2012) (for a review see (Maniadakis and Trahanias, 2014)).

While the robots in this study have different spatial cognition, the temporal cognition used by the robots is simple and identical for both robots. The agents in these studies use their internal clocks for counting time, and use the event of their last meeting as the beginning of a duration (as per Heath et al.

(2012a)). This allows the agents to ground temporal words that correspond to the durations associated with "a short time", "a long time", "a little while". Durations such as these are grounded within events associated with the robots behavior. The duration terms can be used to refer to the temporal aspects of episodes.

7.1.4 Study conditions: Innate vs cross-situational learning

The current research compares two frameworks that learn spatial and temporal terms that are grounded in spatial cognition formed from cognitive maps and temporal cognition formed from meeting events and clock time. We compare learning grounded spatial and temporal language through our new Lingodroids framework (the XS Condition) to a control condition (the Innate Condition). The XS Condition uses XSL to resolve the referential uncertainty between space and time. In the Innate Condition links between word and concept are known *a priori* (identical to previous studies Schulz et al. (2011a) and Heath et al. (2012a)).

7.2 Related work

Two distinct groups of work are related to these studies: learning grounded spatial and temporal language, and learning language through XSL. Several studies have looked at learning grounded terms for space and time, in our own group and others, to name spatial prepositions (Steels, 1995, 1999; Roy, 2002a; Cangelosi et al., 2005), route descriptions (Levit and Roy, 2007; Tellex et al., 2011), spatial relations (Spranger et al., 2014), toponyms (Jung and Zelinsky, 2000; Schulz et al., 2011a), landmarks (Spranger, 2012, 2013), durations (Schulz et al., 2011b; Heath et al., 2012a) and event-based time (Steels and Baillie, 2003; Heath et al., 2012b). These studies can be divided into those that ground directly in perception (Steels, 1995, 1999; Roy, 2002a; Spranger et al., 2014) and those that ground in higher-level representations of space and time (Jung and Zelinsky, 2000; Steels and Baillie, 2003; Cangelosi et al., 2005; Levit and Roy, 2007; Schulz et al., 2011a; Tellex et al., 2011; Heath et al., 2012a, 2012b). The latter group are more relevant to these studies since they require intermediate representations between perception and symbols to allow agents to ground symbols in abstractions that go beyond what is directly perceptible (i.e. space as a location in the world instead of a pixel position in a picture). All of these studies mitigate the problem of referential uncertainty by i) learning space or time independently (Steels, 1995; Jung and Zelinsky, 2000; Tellex et al., 2011); ii) specifying space or time with additional language (Schulz et al., 2011b; Heath et al., 2012a, 2012b); iii) referring to discriminants (Steels, 1999; Steels and Baillie, 2003); or iv) using statistics from XSL (Roy, 2002a).

XSL studies can be grouped into those that have meanings arranged in N-dimensional space (Roy, 2002b; Fontanari et al., 2009) and those that are purely combinatorial (Siskind, 1996; Smith et al., 2006; Yu and Smith, 2007). The former group can be described by a framework that incorporates words, dimensions, categories and values into a context (as in Figure 7.2); however, these studies are all either ungrounded, or grounded directly in perception. The latter group are ungrounded, represent meanings as present or not present, and are based on the assumption that meanings can be directly named without categorization of individual instances.

Our current study differs from those above in that we ground spatial and temporal terms in higher-level representations, and also use XSL used to discriminate between categorized terms.

7.3 How to learn a language for space and time

Robot language learning requires appropriate private and social processes and representations including social interactions, shared attention and representations for storing links between words and meanings (Siskind, 1996; Steels, 1999; Vogt, 2002). The following sections describe in detail the implementation of conversations, shared attention, distributed lexicon tables and referential resolution.

7.3.1 Conversations for learning

As with previous Lingodroid studies, conversations are used to develop symbolic languages (Schulz et al., 2011a). Conversations are the Lingodroids' analogue of language games, first suggested by Wittgenstein (Wittgenstein et al., 1958) and later adapted for robot language studies (Steels and Vogt, 1997; Vogt, 2002).

Lingodroid conversations are divided into two categories: conversations for learning and conversations for performing tasks, with the latter often used to test the former. In previous Lingodroid studies, conversations have been used to learn words for toponyms (Schulz et al., 2011a), distances and directions (Schulz et al., 2012), durations (Schulz et al., 2011b; Heath et al., 2012a) and times of day (Heath et al., 2012b).

In all previous Lingodroid studies, the conversation content provided the link to an underlying sensor or representation, for example in a Lingodroids *where-are-we?* conversation, the presence of the "where" indicates that the conversation is spatial. In the current study we introduce a new conversation - *where-in-space-time-are-we?* and re-use a game from previous studies - *meet-at.*

7.3.2 *where-in-space-time-are-we?*

The conversation *where-in-space-time-are-we* asks a question that allows both spatial and temporal answers. This conversation requires the Lingodroids to associate a word with all possible meanings (see Figure 7.4).

The *where-in-space-time-are-we* conversation is implemented as a question and response in the following steps:

1. The robots establish shared attention.

2. The robots decide on a speaker (the first robot to utter "hello").

3. The speaker asks "*where in space-time are we?*"

4. The listener responds with 1-2 words which can include a label for the current place and a label for the current time in no particular order. If the listener only has a word for the current place or the current time, then they will only provide one word.

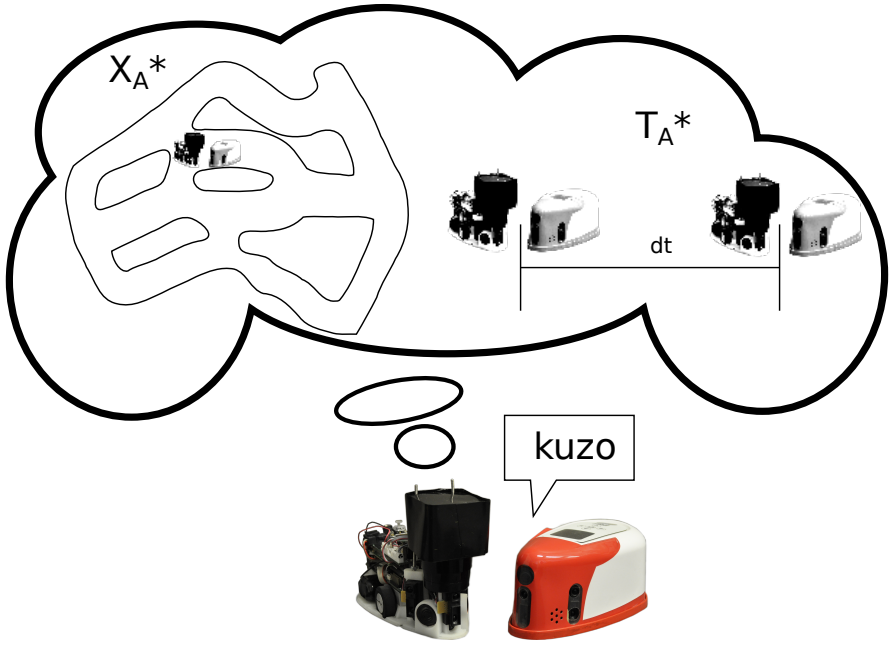


Figure 7.4: The *where-in-space-time-are-we* conversation. A word is provided by a speaker which is then associated with both a location in space (X_A^*) and a duration (T_A^*) by the other robot.

5. If the listener has no words for either place or time, they remain silent and the speaker adds 1-2 words which include a label for the current place and/or time. If the speaker has no words, the speaker will invent two words, one for space and one for time.

6. The listener acknowledges the speaker's words.

7. The two robots check the conversation for errors (there are built-in checks to ensure that the words *"hello"*, *"where"* or *"OK"* are not used as labels).

8. If the words sent and received pass the built-in checks, both robots associate the words heard with the context of the conversation. The robot that provided the words associates them with their corresponding meanings. The other robot associates each of the words with its joint representation of the event location and duration.

7.3.3 Testing coherence: *meet-at*

The *meet-at* game was first introduced in previous work as a practical test for language usage (Heath et al., 2012a). Robots use *meet-at* to specify a future time and place to meet. No associations are formed during a *meet-at* game, and it is used in these studies purely as a test for the *where-in-space-time-are-we* conversation (see Figure 7.5). The conversation is implemented as follows:

1. The robots decide on a speaker and listener.

2. The speaker tells the listener to "meet at" and names a place X_B and a time T_B. The robots independently plan paths to the meeting location and calculate how long it will take (T_T).

3. Both robots wait until just before the time of the meeting ($T_B - T_T$).

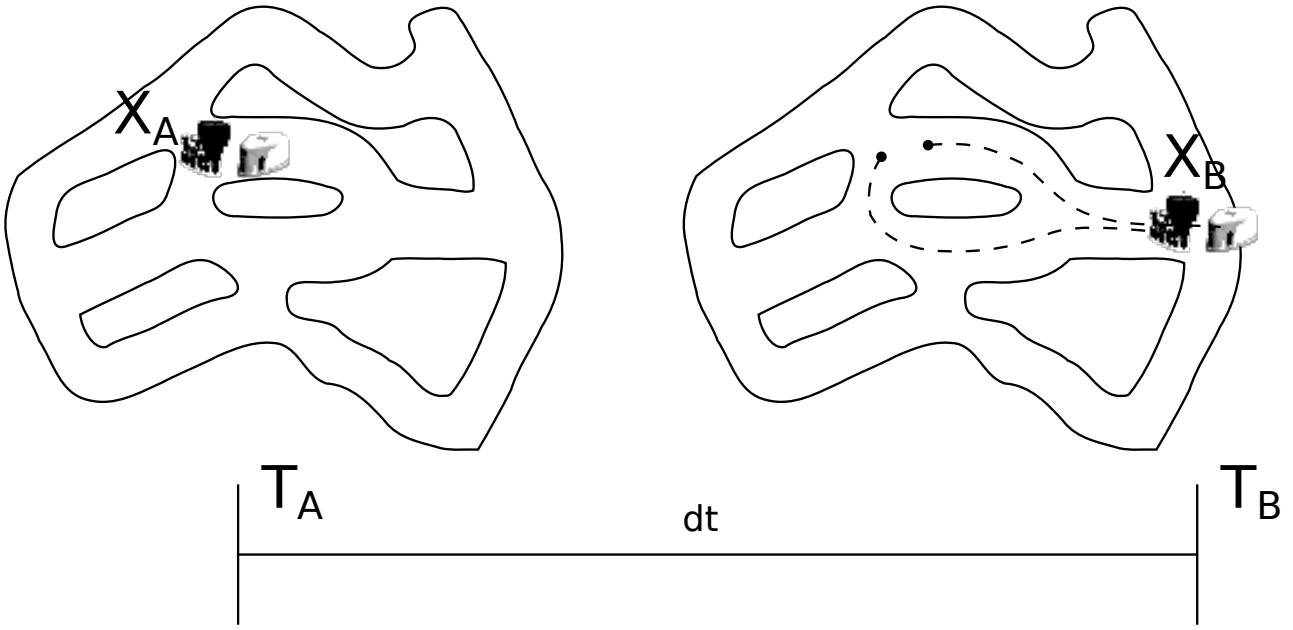


Figure 7.5: The *meet-at* conversation - the two robots meet at place X_A and time T_A and arrange a future meeting for place X_B and time T_B.

4. Both robots move to the location of the meeting.

7.3.4 Shared attention

Shared attention is a requirement for conversations, and to fulfill this requirement *shared experiences* have been used in previous Lingodroids studies (Schulz et al., 2011a). In a shared experience, two agents have internal representations that correspond to the same referent at the same time. Sharing an experience can relate to being in the same place, being in the same time or having some aspect of mental state the same. In previous Lingodroids studies, a shared experience allowed one agent to name an aspect of that experience; however, the challenge is then for the other agent to work out which aspect is referred to. Attention to the aspect of the experience was determined by the type of conversation, *where-are-we?* or *what-time-is-it?* In this study shared attention can be directed toward the current place or current time so referential uncertainty remains, and there are multiple candidate symbol-referent associations during each conversation.

In earlier Lingodroid studies, with bigger environments, shared experience was established by the hearing-distance of two robots - that is the distance over which one robot could hear an utterance from the other (Schulz et al., 2011a; Schulz et al., 2011). In more recent Lingodroids studies using the iRat robots, an overhead tracking camera was used to establish shared attention based on proximity (Heath et al., 2012a; Heath et al., 2013). The overhead camera provided robots with an entering and exiting signal designating co-location. The simulator uses the same method, and shared attention is established by providing a simulated entering and exiting signal depending on the robots' proximities to one and other.

7.3.5 Distributed lexicon tables

The core of Lingodroids is the distributed lexicon table, which is a matrix that stores many-to-many relationships between words and concept elements (Schulz et al., 2011a). The matrix grows dynamically as words and concept elements are added. A key feature of the distributed lexicon table is the separation between the evidence for a concept and the just-in-time concept use. All the equations given in this section are from previous Lingodroid studies (see one of Schulz et al. (2011a), Schulz et al. (2011b) or Schulz et al. (2011) for more details).

An association between element i and word j is incremented (as per step 8 of the *where-in-space-time-are-we* conversation), by setting $a_{ij}{}^* = a_{ij} + 1$, where a_{ij} is the previous association of concept element i and word j and $a_{ij}{}^*$ is the updated association.

For word production, agents find the word with the highest confidence for the feature that they are trying to name. The confidence value h_{ij} for concept element i and word j is as follows:

$$h_{ij} = \frac{\sum_{m=1}^{Y} \frac{a_{mj}(D - \text{DIST-BETWEEN}(i,\, m))}{D}}{\sum_{n=1}^{N} a_{nj}}, \tag{7.1}$$

where D is the neighborhood size - a constant that defines the maximum distance that a word may be generalized, Y is the number of concept elements in the neighborhood of element i, N is the total number of concept elements, a_{ij} is the association between element i and word j and DIST-BETWEEN $(i,\, m)$ is the distance between concept element i and m, that is calculated using Euclidean distance. In these studies the constant D is fixed to different values for different dimensions and robots (see Table 7.1). It is easier for two robots to agree on a term when their neighborhood size refers to a similar generalization distance in environment coordinates. These constants are set to similar values in these studies but could be calculated as a percentage of the perceivable environment.

Words are invented with a probability based on the confidence of the best word given by:

$$p = k \exp\left(\frac{-h_{ij}}{(1 - h_{ij})\, T}\right), \tag{7.2}$$

where $k = 1$, h_{ij} is the confidence of the best concept element for a given word, T is the temperature - an adjustable learning rate that was decreased linearly from 0.1 to 0.0 during the first 100 conversations of the study. The probability p defines an exponential drop from $p = 1$, when $h_{ij} = 0$ to $p = 0$ when $h_{ij} = 1$. The learning rate T adjusts how fast the drop is.

Word comprehension is performed as a neighborhood search across all concept elements, associated with a word. The concept-element with the highest confidence is selected.

7.3.6 Referential resolution

A Bayesian version of Kullback-Leibler (KL) divergence is used as the metric for deciding how confidently a word is associated with a sensor or representation, given by:

Table 7.1: Constants used in production and comprehension

Constant	Value	Description
D	0.25m, 70px, 30secs	Neighborhood distance – distance that the robots generalize
T	0.1-0.0 over course of study	Learning rate temperature – affects the likelihood of inventing a new word
q	$D/3$	Smoothing prior for KDE – q is set to $D/3$ so that the Gaussian kernel formed has around the same shape as the Lingodroids generalization
H_k	3bits	Information constant – how informative a word needs to be to raise its confidence
N_k	6 uses	Word use constant – how many times a word needs to be used to raise its confidence
H_{WEAK}, N_{WEAK}, R_{WEAK}	1bit, 6 uses, 0.5 ratio	These three *weak* constants are the thresholds for when a word should be considered usable. A word needs: $$\begin{aligned} H(w_j, X) &\geq H_{\text{WEAK}}, \\ N_j &\geq N_{\text{WEAK}}, \\ \frac{H(w_j, X)}{\sum_m H(w_m, X)} &\geq R_{\text{WEAK}}, \end{aligned}$$ for word w_j, sensor or representation X, word usage N_j, word count m, and information gain $H(w_j, X)$.

$$p(x|w_i) = \frac{p(w_i|x)p(x)}{p(w_i)}, \tag{7.3}$$

where $x \in X$ are the set of values associated with a sensor or sensory representation, and w_i is a word. The association between w_i and x that provides the most information is given by the KL divergence between the prior and posterior.

Following a similar methodology to Roy (2002b), the confidence H between concept elements relating to a particular sensor or representation, X, and a word w_i, is as follows:

$$\begin{aligned} H\left(w_i, X\right) &= D_{KL}\left(p(x|w_i) \middle\| p(x)\right) \\ &= \sum_x p(x|w_i) \cdot log\left(\frac{p(x|w_i)}{p(x)}\right), \end{aligned} \tag{7.4}$$

where $x \in X$ are the set of sensor or representation data, $p(x)$ is the distribution over representation X and w_i is the event of word i uttered.

To calculate D_{KL} from the Lingodroids' lexicons, a Kernel Density Estimator (KDE) is used:

$$p(x) = \frac{1}{nh} \sum_{i=1}^{n} K_{i,h}(x), \tag{7.5}$$

with the kernel set to a multivariate Gaussian:

$$K_{i,q}(x) = \frac{1}{(2\pi)^{d/2} q^d det(S)^{1/2}} e^{\left(\frac{(x-x_i)^T S^{-1}(x-x_i)}{2q^2}\right)}, \tag{7.6}$$

where d is the number of dimensions of x, S is the covariance matrix, set to the identity for this study (indicating independence between dimensions) and q is a smoothing factor. q was set to $D/3$ using D as the threshold distance used in the Lingodroids' lexicons and the factor of three introduced as 99.7% of probabilities lie within three standard deviations of the mean.

$H(w_i, X)$ is the amount of information (in bits) hearing word w_i provides about representation X. $H(w_i, X)$ is incorporated into the Lingodroid confidence equation, $C(w_j, x_i)$, as an extra term:

$$C(w_j, x_i) = h_{ij} \cdot \frac{H(w_j, X)}{H_k} \cdot \frac{H(w_j, X)}{\sum_m H(w_m, X)} \cdot \frac{N_j}{N_k}, \tag{7.7}$$

where constants H_k and N_k are added to control the number of bits, and number of occurrences respectively that a word needs in order to increase its confidence (see Table 7.1).

There is also a set of weaker thresholds for when a word should be considered usable (described in Table 7.1). The constants in eq. 7.7 and Table 7.1 could be optimized over many datasets, but they are fixed in this study.

7.4 Experimental setup

The studies presented in this chapter were performed in simulation using a modified version of Stage - a kinematic simulator for mobile robots (Vaughan, 2008). We modified Stage to load and render meshes, allowing for a richer simulated camera. The simulator was set up with the environment and robots used in the previous Lingodroids' cross-cognitive capabilities study (Heath et al., 2013). In that study our lab's research robot - the iRat - was used as the robot platform base (Ball et al., 2010). The iRat (Figure 7.6a) is a small rat-animat designed for rat-robot social interaction studies (Wiles et al., 2012). The iRat has also been used for telerobot studies (Heath et al., 2011) and spiking neural networks (Wiles et al., 2010). In all the studies, the iRat's small size allows for small-area studies to be performed with maintained environment detail.

The simulated robots used in this study both have the same motors with encoders and three Infra-Red (IR) sensors allowing them to have identical wall-following behaviors, as per the real iRat. Again, like the real iRat, Robot Operating System (ROS), a robot middle-ware solution, was used to provide the same interface to the robots' sensors in the simulator (Quigley et al., 2009). However, as in Heath et al. (2013), although the two robots both have the base of an iRat, they also have fundamental sensory and cognitive differences. One robot (the iRat) is fitted with an omni-directional camera as the primary spatial sensor

Figure 7.6: The robot platforms and environment - a) the real robots with the laser-bot on the left and the iRat on the right, b) the simulated robots are represented in the simulator as 3D boxes, c) the real Australia maze environment, d) the simulated Australia maze and e) and f) the real and simulated view of the iRat in its environment. The image at the top is the raw camera image that the robot sees. There are two gray scale images under that. The bottom image is a *visual template* - an image preprocessed for matching and storage. The gray scale image in the middle is the nearest matching visual template in the iRat's memory.

and perceives the world as 416×240 pixel RGB images. The omni-directional camera allows vision-based SLAM systems to link views from different angles. The other robot (the laserbot) is fitted with a Hokuyo

laser scanner, allowing a 2D distance-based perception of the world. The iRat runs OpenRatSLAM, an open-source version of RatSLAM designed for ROS (Ball et al., 2013), as its spatial cognition, producing a topological graph. RatSLAM then continues to update the map during the course of the study (Note: learned terms are grounded in the evolving map; however, the map is not changing its representation in response to the terms c.f. Boroditsky (2001) and Walter et al. (2013)). The laserbot first runs Gmapping (also available as part of ROS) to produce an occupancy grid as output and then AMCL to allow it to localize using the occupancy grid (Fox et al., 1999).

The simulated environment (Figure 7.6d) is modeled on the Australia maze, a detailed, feature-rich, environment for the iRats that was used in Heath et al. (2013) (see Figure 7.6c).

7.5 Study – Cross-situational learning for robots with different cognitive capabilities

7.5.1 Aims

The aim of this study is to determine how Lingodroids without *a priori* specification of dimensions can learn spatial and temporal language using XSL (the XS Condition) and compare it to language learning with dimensions given *a priori* (the Innate Condition). The learning algorithms are compared on short-term usability, long-term coherence and time taken. It was expected that usability of the learned language, coherence of the learned language and time taken would all degrade without the prior (i.e. using XSL). The interesting question is the degree of impairment and whether, over time, coherence could be achieved.

7.5.2 Methods

The two simulated robots first mapped the environment with their different SLAM systems. The simulator was then used to produce a large dataset of the two robots' movements and sensors. A single set of 350 (sequential) meetings between the two robots was taken from the dataset, including the locations, times and features available to each agent. This data was then used for Lingodroids' *where-in-space-time-are-we* conversations for the XS Condition and *where-are-we* and *when-did-we-last-meet* conversations for the Innate Condition (see Table 7.2). Short-term usability was analyzed by inspecting the change in language over time. Long-term coherence was analyzed by simulated *meet-at* games: the robots' spatial and temporal representations were transformed back into the simulated environment and the Euclidean distance between the two robots' word centroids were compared within the environment. This comparison is equivalent to the theoretical error that would be present in a *meet-at* game from using the learned lexicons. The time taken for learning was analyzed by looking at the coverage of the environment by converged words over time. Stability of the Lingodroids algorithms were analyzed by running 100 trials using the same set of 350 meetings. For each trial, the term creation, term names and the choice of first speaker and listener roles were randomized.

Table 7.2: Innate Condition vs XS Condition

Process	Innate Condition	XS condition
Conversations for learning	*where-are-we?* *when-did-we-last-meet?*	*where-in-space-time-are-we?*
Conversations for testing	*meet-at*	*meet-at*
Learning type	One-shot association	XSL
Lexicon	Separate lexicons for space (experiences) and time (durations)	Same lexicon for space (experiences) and time (durations)
Production rule	$\text{argmax}_j\,(h_{ij})$ (for concept-element i and word j, see Equation 7.1)	$\text{argmax}_j C\,(w_j, x_i)$ (for concept-element i and word j, see Equation 7.7)
Representations	RatSLAM and Gmapping maps, clock time	RatSLAM and Gmapping maps, clock time
Expected advantages	*A priori* information leads to shorter learning time and increased immediate usability	Agents autonomously learn without *a priori* information, but take longer

7.5.3 Results for a single trial

The robots in the XS Condition created a total of 45 words during the study with 29 words that were usable by at least one of the robots at the end of the study. Of the 29 converged words, 14 had a higher information gain ($H(w_i, x)$ - see Equation 7.4) for location, and 15 a higher information gain for duration. The robots in the Innate Condition created a total of 38 words during the study with 36 in effective use after the study. The words from the Innate Condition were divided into 19 location words and 17 temporal words. A higher number of unused words were created in the XS Condition (16), because several words were generalized so far that they ceased to be considered informative by the robots. Otherwise the word counts were similar, with four more location words and two more temporal words created in the Innate Condition.

Each word is associated with a set of concept-elements, which are points in space and time where conversations were held (stored in each robots' individual distributed lexicon tables, as described in Section 7.3.5). Different views of a word's meaning are possible by looking at either i) the concept elements - which give an indication of a word's meaning within space and time; and, for the XS Condition, ii) the information gains in space and time - which give an indication of whether a word refers to space or time. Words can also be viewed from a comprehension or production perspective. Comprehension is based purely on the generalization of the concept elements, while production additionally takes into account the competition of other words in the language.

The spatio-temporal language that results from the XS Condition represents meanings as confidences that encompass regions in space and time (see Figure 7.7 for the locations and durations *comprehended* by the robot). For *production*, the different words compete against each other, with an agent choosing the word with maximum confidence for a given feature.

The information gain was calculated at each time step of the experiment (see Figure 7.7, col 3). These

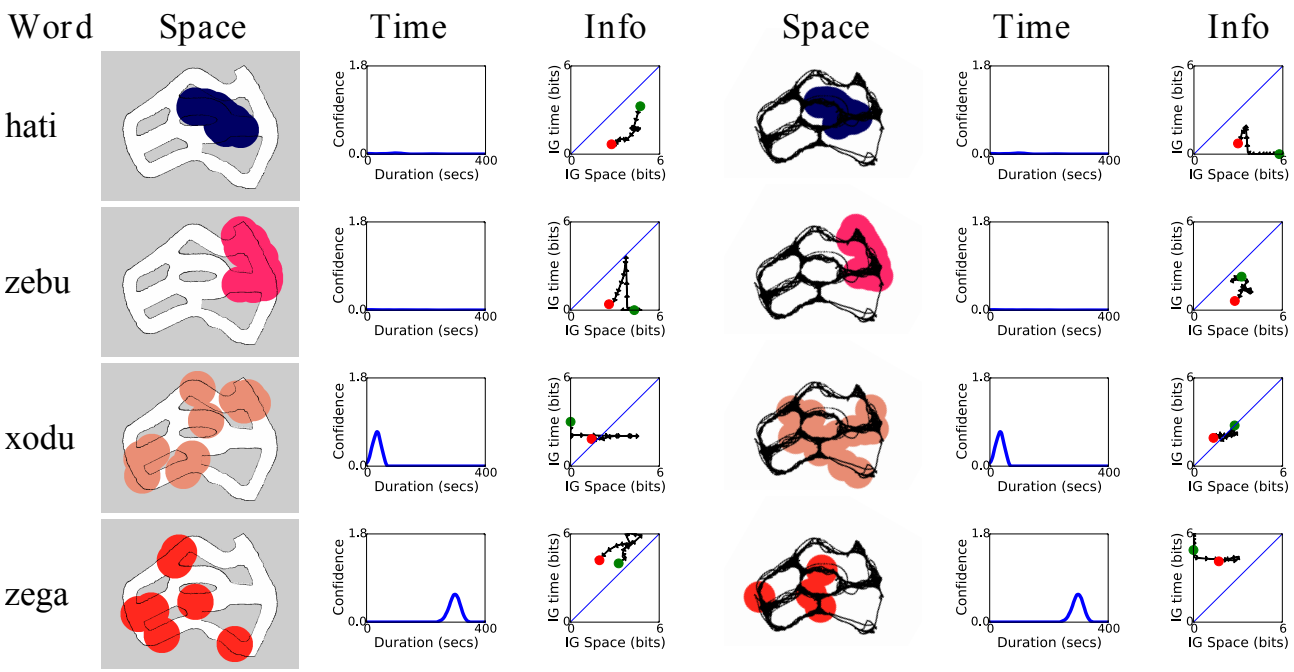


Figure 7.7: Words for space and time learned by the Lingodroids in the XS Condition. Laserbot representations (left); iRat representations (right); word confidences for locations in space (col 1) and durations in time (col 2); trajectories of the word over time in information space (col 3); temporal information gain (vertical axis); spatial information gain (horizontal axis); equal temporal and spatial information gains (blue diagonal). Arrows indicate word journey initial (green) to final (red) information gains. The robot that invents the word has its green dot on the axis. The diagonal partitions information gains with space above and time below the diagonal. Note that trajectories converge to similar final information gains for both robots.

trajectories show the journey of a word's representation during the experiment. Words are generally created with the highest possible information in both space and time, and over multiple experiences, the information is refined until the word provides information solely about either space or time. It is important with such dynamics that a decision to link a word to time or space is delayed until enough conversations have been held.

The short-term usability can be expressed in information space. Whereas the dimension of a word formed under the Innate Condition is immediately known by both agents, a new word cannot be used in the XS Condition until its usage has converged with high confidence.

Hence, representations for a word must be different for a speaker and listener the first time a word is used. The speaker must be willing to immediately use the word again to further establish its meaning; however, the listener should not use the word until they can confidently associate the word with a dimension.

Long-term coherence is evaluated by looking at the robots' ability to interpret words consistently to the same places and times in the environment. The robots' calculated arrival times and locations for the XS Condition differ by 4.5s (stddev 6.3s) and 0.17m (stddev 0.16m); and for the Innate Condition 11.9s (stddev 9.4s) and 0.12m (stddev 0.06m). Within this study, the coherence for the XS Condition is better for time, whereas the coherence of Innate Condition is better for space. As the environment is 2.5×1.8m, a spatial term would allow the robots in the XS Condition to restrict the environment to 2% of the area

and robots in the Innate Condition to restrict to 1% of the are (assuming a circular area for the error). As the conversations cover 400 seconds, a temporal term allows the robots in the XS Condition to restrict a duration to 2.2% of the possible durations and robots in the Innate Condition to restrict a duration to 4.7% of possible durations.

Language learning was evaluated as coverage of the environment by converged words over time (see Figure 7.8). For the iRat, the spatial coverage was given by the percentage of experiences (nodes in the map) that were described by a word. For the laserbot, the spatial coverage was given by the number of free grid squares that were described by a word. The iRat's coverage was biased towards where it had been more often, so its coverage appears better than that of the laserbot for both learning algorithms. The laserbot did not completely learn the free grid squares with either algorithm, as there was a small region of the map that was not visited at all.

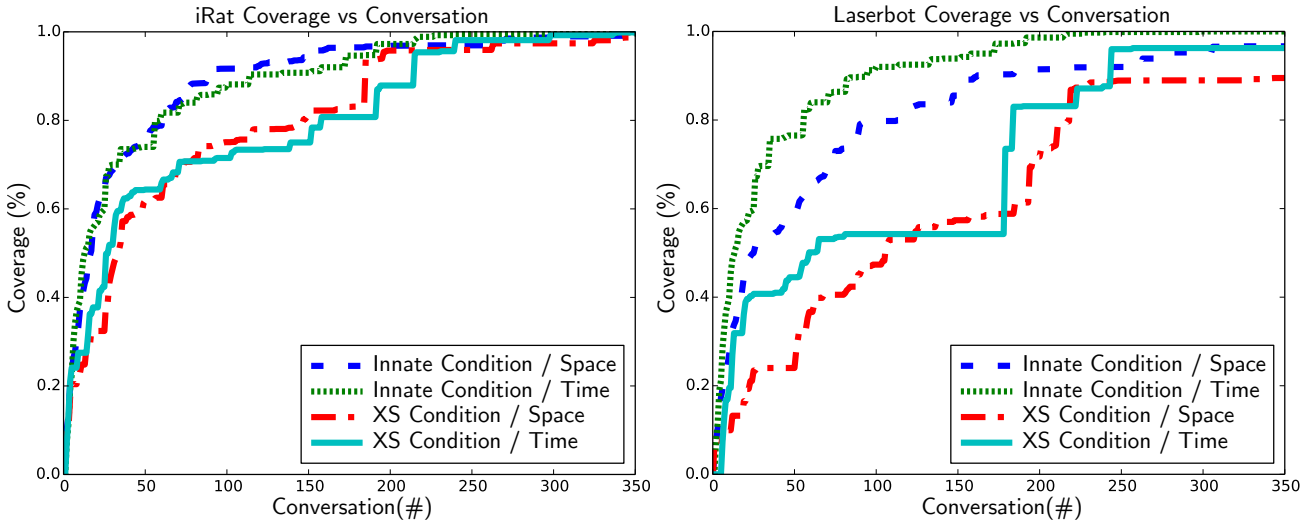


Figure 7.8: The coverage of learned language over time. The coverage is defined as the percentage of experiences that the iRat can name and the percentage of free grid squares that the laserbot can name. a) the coverage of the iRat and b) the coverage of the Laserbot. Both plots show coverage for space and time for both the Innate Condition and the XS Condition. The Innate Condition requires a significantly shorter time for coverage of the environment.

The temporal coverage for both robots was calculated by the percentage of durations between 0-400s that the robots could name. Again the laserbot in the XS Condition did not reach 100% coverage of durations, probably due to the rarity of long durations in conversations.

The key points in these results are that the Innate Condition had significantly faster learning time than the XS Condition, but given sufficient time, the lexicons in the XS Condition reached the same levels of coverage.

7.5.4 Results for 100 trials

To evaluate the stability of the algorithms used in the two study conditions, the same set of 350 meetings was run over 100 trials. For the XS Condition, the average distances between the robots' terms were 0.14m (stddev 0.16m) and 10.11s (stddev 38.98s), and for the Innate Condition, 0.11m (stddev 0.067m)

and 12.56s (stddev 19.79s). The higher standard deviation for temporal terms in the XS Condition was due to rare cases where agents incorrectly associated a term with space or time. This occurred when an infrequently sampled part of space or time was captured by the use of a term that had become common for the other dimension. The high information provided by the association of an infrequently sampled part of a dimension linked the term to that (incorrect) dimension. Although only 0.48% of the total terms, the average and standard deviation were markedly affected. The average for the XS Condition without these terms was 6.75s, stddev of 9.85s (c.f. XS Condition temporal terms above). For the Innate Condition, two outliers were caused by incorrect associations due to shared attention mismatches. The 100 trials showed that while the algorithm was stable for the majority of cases, there were rare cases where the robots disagreed on a word's informativeness.

7.6 General discussion

This study demonstrates how referential uncertainty about spatial and temporal dimensions can be resolved for higher-level grounding. Spatial and temporal terms are grounded in cognition, and information gains are used to resolve terms for space from those for time.

Our aim in this chapter was to analyze the effects of using XSL on language learning performance. We expected that the major advantage of XSL would be flexible learning that could be applied to different sensors and cognition with minimal innate assumptions; but, both language usability and learning times would be degraded. This study shows that with respect to usability, only immediate usability was affected, and long-term coherence reached similar levels in both the XS and Innate Conditions. Learning time was increased as expected with the robots in the learning condition taking longer to achieve convergence to a dimension.

The current process would benefit from further refinement. An unexpected phenomenon was revealed by the information space analysis: the information for the incorrect dimension "spikes" for a word inventor when the other robot says the word back to them for the first time. This spike is due to both robots always using cross-situational learning no matter how sure they are of a word's meaning. Removing this constraint (i.e. allowing a confident robot to completely link a word to its dimension) would also remove the spike; however, words could no longer change meanings.

The large differences in the temporal terms is surprising for both conditions, as computer clocks are very accurate. The reason is that the time at which the agents associate a word is different for the speaker and listener, lengthening or shortening the temporal referent for one of the agents. This issue could be addressed by using the shared attention signals to ground the beginning and end of the duration.

7.6.1 Design choices

The implementation of a symbol grounding framework requires a set of design choices that affect flexibility of learning (innate vs what can be learned), learning time, computational and memory requirements. The following sections describe these design choices.

Spatial and temporal cognition: Spatial and temporal cognition allow a robot to have better representations of space and time, based on sequences of perception. It is possible to ground spatial terms directly in perception as has been done within scene description tasks (Steels, 1999; Roy, 2002a); however these studies are limited to a single egocentric view.

For mobile robots capable of locomotion within their environment, allocentric representations of space provide location, pose and spatial competencies for movement planning. SLAM systems are the state-of-the-art mapping systems for mobile robots, providing them with the capability to learn allocentric spatial representations. Symbol grounding within SLAM systems allows robots to name features of advanced spatial representations (Jung and Zelinsky, 2000; Schulz et al., 2011a; Walter et al., 2013). For spatial grounding, SLAM representations allow agents to name places other than the "here and now" (Schulz et al., 2012) and paths through the environment (Tellex et al., 2011).

In the current study, the agents' spatial cognition was implemented as SLAM representations through topological and occupancy grid maps. These representations allow grounded labeling of toponyms. This study used both static and dynamic maps. Gmapping was used by one robot to create a static map, so that AMCL could be used to localize within the map throughout the study. RatSLAM was used by the other robot to create a map at the beginning of the study and then the map was corrected throughout the study. The Lingodroids-RatSLAM pairing allows the meanings of words to dynamically change according to corrections of the map; however, unlike in Walter et al. (2013), language does not affect the learned map.

It is not possible to ground temporal terms directly in perception, so previous studies have used sequences of perception to ground terms for event descriptions (Steels and Baillie, 2003), sequencing (De Beule, 2006), durations (Schulz et al., 2011b; Heath et al., 2012a) and cyclic events (Heath et al., 2012b).

Temporal cognition in the current study is simple, relying on only the agent's hardware clocks and an event shared by the agents (a meeting). However, even this simple representation is enough for the agents to label durations.

Although spatial and temporal cognition can be dynamic models, innate processes underlie the spatial and temporal cognition in all symbol grounding studies. It may be technically possible to bootstrap spatial and temporal cognition from learning or language, but we note that animals develop competent spatial cognition without language.

Shared attention: Shared attention is critical for symbol grounding so that agents are able to learn shared words to describe the same referents. The current study uses a simulated proximity sensor to establish shared attention. Other methods for establishing shared attention include pointing (Steels et al., 2007; Spranger et al., 2014), recognition of a fiducial marker (Vogt, 2002), hearing distance (Schulz et al., 2011a) and overhead camera (which functions as a proximity sensor) (Heath et al., 2012a). Important characteristics of different methods include the granularity that the robots can attend to (pointing allows robots to be more distinct than hearing distance), the number of referents in a shared context and the affordances a robot needs. The number of referents in a shared context affects the referential uncertainty that is associated with that context (Smith et al., 2006). In the current study, conversations in the Innate Condition limited the size of the context to one referent, whereas the XS Condition allowed up to two.

An open question in XSL is whether it scales to larger languages (number of words, dimension and referents). Smith et al. (2006) suggest that when the number of dimensions approaches the number of

categories the learning time can become unfeasibly large; however, this computational complexity can be mitigated by exploiting distributions of named referents (Blythe et al., 2010) or by introducing additional social conventions.

The use of a shared experience in this study was critical to the ability to refer to time. While pointing can be used by robots to name objects (Steels et al., 2007), it is not possible to point at time. Robots must instead refer to time through language as in the Innate Condition (and previous work (Heath et al., 2012a)), or allow temporal referents to be present in the context and resolve the temporal dimension, as in the XS Condition.

Conversation mechanics: Conversation mechanics provide coordination between communicating agents. They are typically implemented as finite state machines (Steels, 1999; Schulz et al., 2011a).

An advantage of using conversations is that studies are easily adaptable to more than two agents using an iterated learning model (Kirby and Hurford, 2002). This design choice allows language learning to be easily extended to populations (Schulz et al., 2012).

An ongoing issue related to conversations is corrective feedback (Steels, 1999), which is usually expected to make learning faster. However, Fontanari and Cangelosi showed that the gains from corrective feedback were mainly short term (Fontanari and Cangelosi, 2011), and early infant language learning does not require corrective feedback (Bloom, 2002).

On-line learning: On-line learning, the ability to learn during usage, requires: i) that at any point in time during a study the language status can be evaluated, and ii) that agents rely on sensory data available to them during the study. It is typical for conversation-based studies to use on-line learning (Steels, 1999; Vogt, 2002). Off-line studies use multiple sources of data, including some which are not available to the robot during testing, and often require multiple presentations of the same inputs (Fontanari et al., 2009; Tellex et al., 2011).

Both conditions in the current study learn on-line; however, the results for the immediate usability of a word in the XS Condition (particularly based on Fig. 7.7, cols 3 and 6) show an important difference between the two conditions. Whereas robots in the Innate Condition immediately assigned a word to a dimension (one-shot learning), robots in the XS Condition needed several conversations to allow words to converge to a higher information gain. Both conditions only required one presentation of each different referent during learning (c.f. Fontanari et al. (2009)).

Representations of links between words and meanings: Symbol grounding studies need to store links between words and meanings. Representations for links between words and meanings come in two types: those that treat words separately from sensory data (Steels, 1999; Roy, 2002b; Schulz et al., 2011a), and those that treat words as another sensory input (Fontanari et al., 2009). The advantages of the former are that words can be linked directly to any categorization algorithms, or to exemplars (as they are in the current study). There is also no need to transform the words into a signal space. The advantages of the latter are that symbols can be treated like other sensory input, and clustering can be related to words, or media that makes up the word, such as audio or light.

Both conditions in the current study used lexicon tables, but in different ways. The Innate Condition used a separate lexicon table for spatial terms and temporal terms, whereas the XS Condition allowed links to both spatial and temporal features within the same lexicon table. Important criteria for using the lexicon table for XSL and referential resolution are the many-to-many relationships between words and exemplars that are possible within the Lingodroids distributed lexicon tables. This type of relationship allows the lexicons to store multiple contexts simultaneously under the same word.

Referential resolution: The two learning conditions in this study demonstrated how conversations could resolve referential uncertainty by either i) using innate terms *when* and *where*, or ii) extracting regularities across multiple conversations. For discrete features, rule-based and usage counts can be used to extract regularities (Siskind, 1996; Smith et al., 2006). However, when features are continuous, the equivalent metric is information (Roy, 2002a).

The current study uses KL-divergence conditioned on space and time to link words to a single dimension. Such linking provides the opportunity to name that dimension in future studies.

Previous work has demonstrated how KL-divergence can be extended to multiple dimensions by forming categories from greedily integrating combinations of values (Roy, 2002a), allowing a word to refer to both a place and a time.

Siskind makes a key point about statistical learning: the number of times a word and meaning co-occur may be outnumbered by the times that they do not occur (Siskind, 1996). This observation underlies why the current study and others (Roy, 2002a) use KL divergence instead of mutual information.

Categorization and generalization: Studies that ground in robot sensors need to be able to cluster similar sensory readings into a single category. Naming every previous sensory reading creates terms that can not generalize to future sensory readings.

There are many different ways of categorizing, and previous studies have used Gaussians (Roy, 2002a; Fontanari et al., 2009), discrimination trees (Steels, 1999), neural networks (Tikhanoff et al., 2011) and non-parametric distributions (i.e. where a distribution is formed directly from previous collected data)(Schulz et al., 2011).

Due to the on-line nature of Lingodroid studies, the timing of categorization (i.e. when categorization occurs) is a key factor. Lingodroid studies use a just-in-time dynamic process for categorization in both production and comprehension. This process is based on a non-parametric representation, which allows a distribution to be evaluated dynamically.

Confidence metrics: A confidence value for a word and feature plays an important role in language production where it is used to allow competition between words when attempting to name a feature (Steels, 1999; Roy, 2002b; Vogt, 2002). In the current study, confidence is calculated in three ways: i) a comparison between category and feature; ii) KL-divergence between and the dimension of the feature; and iii) the word usage count. Restricting the confidence based on the word usage count increased the probability of a word maintaining its originally intended meaning. Without this restriction, one agent can immediately change the meaning of a word by overturning the other agent's definition. The current implementation allows for meaning negotiation.

In a small number of cases, a common word that was accidentally used with rare occurrence could take on the rare meaning and reject its original meaning. An example from natural language that mirrors this phenomenon might be when hearing a word, a rare event occurs, such as a rabbit running past, but instead of naming the rabbit, the speaker looks at the sun and says "it's late". The rarity of the rabbit running past makes the phrase "it's late" very informative about the rabbit compared with it being late in the day Quine (1960).

Social conventions: "Social conventions" are additional constraints on conversations that can be used to improve the learning rate or enforce conditions. One important social convention is used in these studies: if a robot already has a (suspected) word for a place or duration, the other robot would not provide another word. This convention stops the two agents from creating a shared language where the production maps are completely different (i.e. the agents will understand each other but use different words to refer to the same place or time). This social convention is implemented as an extra step in the *where-in-space-time-are-we* game (step 5). If a speaking robot has no words, then it remains silent during its turn, and the listening robot is then aware that its partner has no words to describe the current context. The listener is then free to either use its own existing words or invent new words to describe the context.

7.6.2 Different cognitive architectures

There are two previous studies of symbol grounding across heterogeneous robots: i) Jung and Zelinsky demonstrated grounding across robots with different sensors, using language-game like interactions, but without actually inventing or generalizing names (Jung and Zelinsky, 2000); and ii) a previous Lingodroids study extended language learning to agents with different sensors and cognition (Heath et al., 2013). The current study differs from both the previous studies as it enables XSL on robots with different cognitive capabilities. For heterogeneous robots which can potentially have many different, unique sensors and representations, the ability to autonomously decide which word belongs to which dimension allows them to correlate the environmental similarities between their different sensors and representations through language.

Steels argues that a language is represented across a community, and that agents can have different understandings for the same word (Steels, 1999). Steels' study and several other conversation studies also analyze language change across populations of agents and the differences between the meanings held by individual agents in the population (Steels, 1999; Kirby and Hurford, 2002; Schulz et al., 2012). This study complements these analyses by exaggerating the differences between the meanings held by two agents.

Robots with different sensors and cognition have much in common with sensor fusion (Khaleghi et al., 2013) and multi-modal learning (Mangin and Oudeyer, 2013) studies, although unlike these fields, the current study socially separates the different representations of space. The different "modalities" learned in the current study refer to the same part of the environment, but like the study of Mangin and Oudeyer (2013), have the ability to (socially) produce one spatial representation from an instance of the other.

The Lingodroids' different representations of space provide an interesting perspective on a word and its meaning. Traditionally, studies within a framework defined by Peirce (Ogden and Richards, 1923) use identical agents, and meanings for different agents converge towards the same representation during

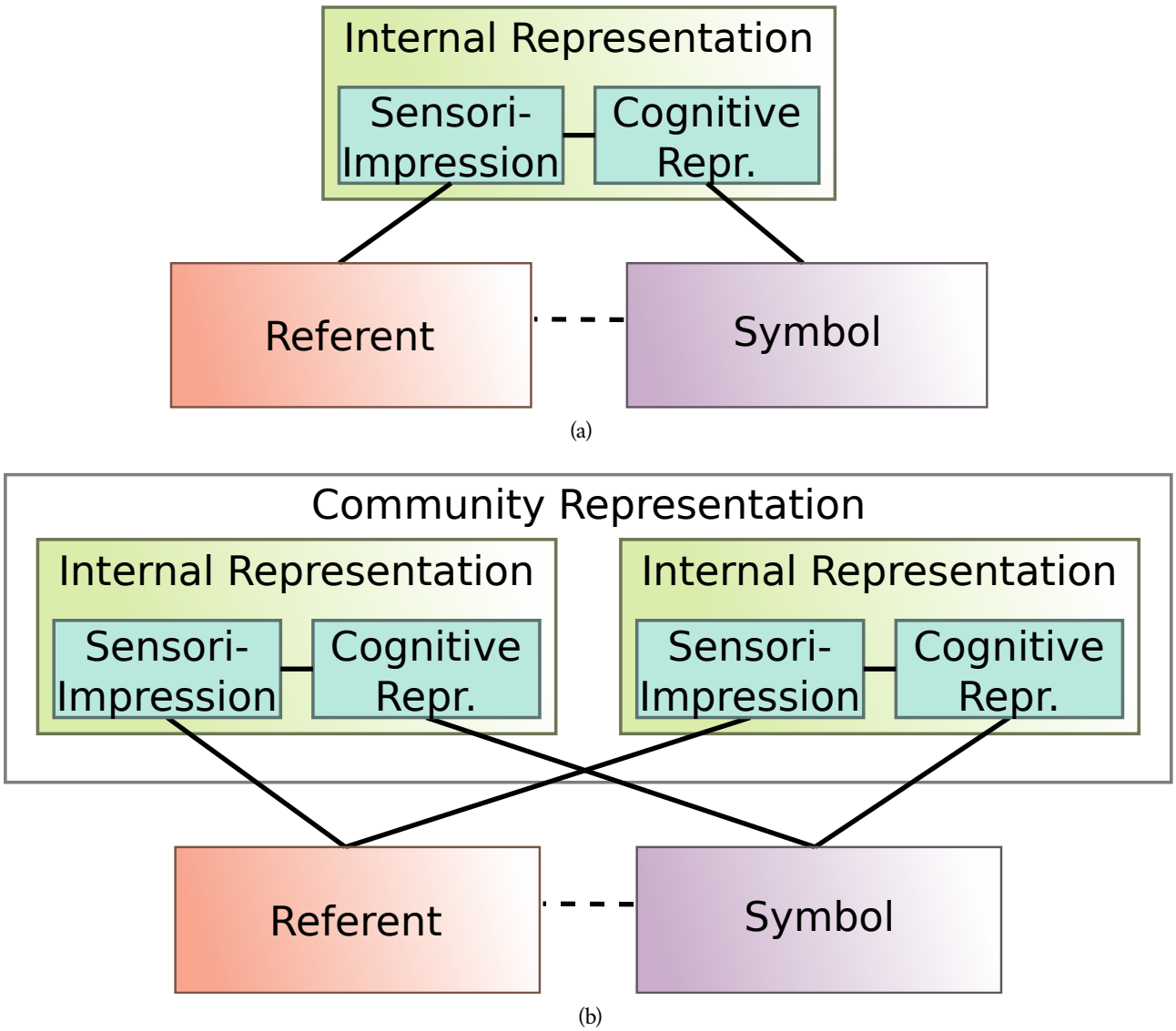


Figure 7.9: Extensions of Peirce's semiotic triangle. a) dividing the internal representation into a sensory impression and cognitive representation, and b) the semiotic triangle across two agents - for successful language learning it is important for the referent and the symbol to be comparable; however, the sensory impression and cognitive representations of two agents can be completely different.

learning. However, although the robots' distributions are always similar, and the underlying information converges, the meaning of the same shared word between two agents is derived from fundamentally different sensory evidence, one based in vision, the other in laser range-finding data. The meanings also depend on different underlying representations of the world, one based in a topological map and the other based in an occupancy grid. These robots demonstrate the extremes possible in community based word representations. We can extend Peirce's semiotic triangle to illustrate a deeper understanding of an individual's (Figure 7.9a) and a community's (Figure 7.9b) representational structure. Importantly, the referent and symbol must be the same for all agents for language learning to be effective.

7.7 Conclusions

In this chapter we have demonstrated that referential uncertainty for the dimensions of space and time can be resolved using XSL. We demonstrated a practical solution using Lingodroids, which was able to create a usable language. We compared the learned language with another language created by building in dimension associations *a priori*, and showed that XSL allows unconstrained learning of the designated meaning for a given word, where links can be flexibly created based on the information a word gives about a cognitive representation. Using XSL the Lingodroids were able to maintain long-term coherence of the learned language, but had the added issues of a delay in immediate usability of a word and extended time taken overall for learning.

Chapter 8

General discussion and conclusions

The goal of this thesis was to develop and analyze a new framework, L2, for language learning to address: i) grounding spatial and temporal terms in spatial and temporal cognition, ii) grounding spatial and temporal terms in different underlying cognitive architectures, and iii) resolving referential uncertainty for spatial and temporal terms. The studies demonstrate a lexicon learning framework that addresses the points above. The framework extends previous Lingodroids studies that were originally designed for spatial language learning (Schulz et al., 2011a), to durations (Study I, Chapter 4), cyclic temporal events (Study II, Chapter 5), different cognitive architectures (Study III, Chapter 6) and XSL (Study IV, Chapter 7). The XSL study adopted features from each of the previous studies into a single, integrated framework.

8.1 Contributions

This thesis (and associated publications) provide several contributions relating to symbol grounding, grounding using mobile robots with different cognitive architectures and referential uncertainty.

Development of representations and conversations for grounding and using terms for durations: The L2 framework allowed the robots to develop simple temporal cognition grounded in durations and use their duration terms to organize meetings (Study I, Chapter 4). The ability for mobile robots to learn and use language grounded in simple temporal cognition provides the foundations for a variety of different tasks including autonomous multi-agent planning, exploration and synchronization.

Previous studies into temporal cognition for robots have only covered events and sequencing (Steels and Baillie, 2003; De Beule, 2006; Schulz et al., 2011b). The approach in this thesis is to take the Lingodroids methodology for learning spatial language and apply it directly to the robots' clock time. However, due to the transience of time (Galton, 2011), an analogue of the Lingodroids toponym is not usable in time. The grounding must instead be immediately be generalized to the future, making durations an obvious candidate.

Previous Lingodroids studies have looked briefly at time, using cognitive maps to ground durations in shared journeys (Schulz et al., 2011b); however, grounding in clock time via meetings is advantageous in that it covers a wide range of times and has minimal noise.

Development of representations and conversations for grounding terms for times of day: Although clock grounded time is possibly the most convenient for a robot, the range of types of time used in natural language exceed that of clock time (Gibson, 1975; Frank, 1992; Engberg-Pedersen, 1999; Kuipers, 2008). Attempting to ground cyclic time using clock time can lead to subtle, but important differences that can affect understanding and planning.

Times of day are both event-based time and cyclic and as such they are used in many languages. Of particular interest to these studies are the Amondawa, who appear to use only event-based time (Sinha et al., 2011). The development of methods for grounding and using times of day allows the development of robots that can learn Amondawa temporal concepts, and use the concepts for organizing meetings.

Development of representations and conversations for grounding across robots with different cognitive architectures: There is only one previous example of symbol grounding across different cognitive architectures for mobile robots (Jung and Zelinsky, 2000) and the robots are cognitively identical and their language learning framework is limited in how features can be labeled and generalized.

The approach in this thesis develops a framework to allow robots with very different spatial sensors and representations to learn spatial lexicons grounded in their own subjective experiences of space. While Jung and Zelinsky studied a task where both robots only required the same spatial cognition, for more complicated tasks, communication grounded in cognitive differences will allow robots to share their own interpretations of their environment. This capability contributes towards mobile robots that can perform complicated tasks with specialist sensors and cognition.

Development of representations and conversations for generative bootstrapping of identical features from different cognitive architectures: Grounding transfer can enable robots to use previously learned categories to bootstrap new higher level categories (Cangelosi and Riga, 2006). In previous Lingodroids studies, grounding transfer was applied to space, through the process of generative grounding - where distances and directions can be learned from sets of toponyms (Schulz et al., 2012). New toponyms can then be learned from previously learned toponyms, distances and directions.

In this thesis, generative grounding was extended to robots with different cognitive architectures, to allow robots that have terms grounded in completely different spatial representations to learn the higher order distances and directions that are grounded identically. This characteristic of the L2 framework allows identically represented terms to be formed from those represented differently, and therefore allows robots to develop a shared social cognition from different private cognition.

Development of methods for choosing the best word to describe a feature: Many studies in the symbol grounding literature provide a process by which a word is chosen to describe a feature (Steels, 1999; Roy, 2002a; Marocco et al., 2010; Schulz et al., 2011a; Tikhanoff et al., 2011); however, none handle a scenario with all the following: i) learning words that are not directly grounded in perception, ii) learning is online (i.e. words must be immediately usable), iii) words are generalized, iv) robots have different cognitive architectures, and v) there is referential uncertainty. For mobile robots, all of these points are important.

The approach in this thesis demonstrates a method that addressed all of these points, and tests the method in the study of Chapter 7, demonstrating that the L2 robots could learn lexicons that could be used for *meet-at* games. Several variables are important in choosing the best word to describe a feature in this scenario. For grounding, confidence for a word and feature pair needs to depend upon: previous uses of the word and feature, and previous uses of the word with *other* features. Referential resolution depends on information provided by the word about the current dimension and information provided by the word about other dimensions. Online usability depends on the number of times the word has been used before. Generalization depends on the distance (or some other metric of the current feature from the previous uses of the word).

8.2 Measuring success

There are many ways of evaluating the performance of a language learning framework (see Section 2.2.5). The lexicons in each of the studies in this thesis were compared using variants of the Lingodroid's coherence metric (Schulz et al., 2011a). Coherence provides a measure of the similarity between two agents' lexicons that is judged by production (see Section 3.5). Evaluating based on production, instead of comprehension, ensured that the agents understood each other and were producing the same words for the same features. Coherence decreases when agents understand each other but produce different words to describe the same feature.

The lexicons of the two agents were compared in the first two studies using coherence, as the identical groundings of the agents in these studies allowed their lexicons to be compared directly. Coherence of above 80% was considered high, and above 70% was considered moderate. Durations and cyclic time were found to have high coherence, as both the robots' clocks (used for grounding durations) and the sunlight signal (used for grounding cyclic time) were noise free and highly accurate. The spatial coherence was moderate due to the noise inherent in the images and odometry inputs to RatSLAM, and the small differences between shared experiences of the two robots (i.e. a successful *where-are-we* conversation requires that the Lingodroids are in the same place, but there are typically small errors). Local spatial coherence was high where the robots met, while the error accumulated the further a word was generalized from local evidence.

For the studies of L2 robots with different cognitive architectures (Studies III-IV, Chapters 6-7), coherence was not easily calculable. Different metrics were used for the study instead: an edit distance of toponyms along a journey indicated similarity between word usage, while the coherence of distance and direction lexicons formed from the robots' toponyms were compared directly, and the coherence was found to be moderate (see Study III, Chapter 6).

Three of the studies featured the *meet-at* language game as a practical way of demonstrating lexicon usability. In the temporal studies (Studies I and II, Chapters 4 and 5), the robots held the meeting and the error was measured for the task as a whole. The L2 framework contains the following components, where quality is difficult to measure:

- the robots' maps, particularly the semi-metric topological map of RatSLAM, which is difficult to compare to a ground truth;

- the robots' lexicons when they have different cognitive architectures; and
- the robots' communication abilities, which can have issues with shared attention and WiFi noise.

The benefit of using *meet-at* was that it measured the the cumulative error from all of the L2 components without needing to assess them individually. The *meet-at* game was used for analysis in the XSL study, but the meetings were not held. Instead the error between the centroids of the spatial and temporal words were measured, which would be the meeting place and time of the two robots if they were to organize a meeting. Across all the studies, the error between word centroids was minimal, as the word centroids tended to be close to the local evidence gathered by the robots. More failures occurred in the practical *meet-at* game, due to the robots' difficulties in navigating within their estimated schedules. Increasing the robots' navigation sensors and improving the robots' navigation abilities would increase the successful trials in these studies.

In summary, learning spatial and temporal language was successful. The coherence was moderate to high in all tests where applicable, and the Lingodroids were able to succeed at the practical *meet-at* game in a majority of cases. The Lingodroids were able to select spatial and temporal terms and have the other robot understand them to within 5% of the environment for both conditions of the XSL study (Study IV, Chapter 7).

8.2.1 Repeatability

The studies in this thesis are intended as proof-of-concept studies, and mapping and language learning were only performed a single time in Studies I-III (Chapters 4-6); however, the testing of the language was performed multiple times for each of these studies. Using real robots and high numbers of conversations meant that all the studies in this thesis took large amounts of time to setup and run, so repeating all facets of the studies was not always achievable. For the XSL study (Study IV, Chapter 7), the mapping and conversations were only performed a single time, but the language learning was run 100 times offline for the same maps and set of conversations and demonstrated that the results of running the learning once were in fact a suitable exemplar for the additional trials. The studies in this thesis are worth repeating. To improve performance the following factors could be improved first to increase successful results: i) improved robot navigation with RatSLAM, and, ii) improved localization using AMCL in Study III (Chapter 6).

8.3 L2 as a general purpose, lexicon learning framework

The L2 framework can be viewed as a general purpose, lexicon learning framework with a unique set of characteristics. The methodology for developing the L2 framework was to start with the core features of the Lingodroids framework – a framework already capable of learning spatial language grounded in cognitive maps (Schulz et al., 2011a) – and add components to address grounding temporal terms, grounding across different cognitive architectures and handling referential uncertainty. Characteristics of the L2 framework come from both the previous Lingodroids incarnation, and also the features added in the studies in this thesis.

8.3.1 Characteristics of the L2 framework

The L2 framework is formed from the core processes and structures from previous Lingodroids studies (the state-of-the-art in spatial language learning) and new capabilities for bootstrapping temporal terms, using additional cognitive spatial systems and performing XSL. New capabilities were added to the L2 framework for each study in order to handle each new scenario. As such, the features added for each study provide insight about the nature of the problems tackled. Characteristics from previous Lingodroids studies and new additions are given in the sections below.

Stable characteristics from previous Lingodroids studies: The L2 framework's conversations, distributed lexicon tables and processes for association, production and generalization were derived from the previous Lingodroids. The successful application of these core data structures and processes to temporal terms, different cognitive architectures and XSL, demonstrate how the Lingodroids core algorithms are robust across a range of scenarios. The previous Lingodroids framework afford L2 the following important core capabilities (detailed in Section 2.4.6): i) private grounding in cognitive maps (spatial cognition), ii) learning through conversations and shared experiences, iii) online learning, iv) non-parametric representations of associations, v) categorization delayed until required for production, vi) grounding transfer using spatial features, and vii) word invention.

Real robots: For studies I-III (Chapters 4-6) real robots were used as the L2 framework's robot platforms. The L2 framework used the iRat robot platform, and also a modified iRat called a laserbot. Using real-robots for language learning instead of simulators presents the following challenges:

- real-world noise – the noise that is present in a robot's sensors from the real-world can be difficult to capture within a simulator;
- construction and/or maintenance of the robot platform – the robot must be reliable and parts repaired or replaced when necessary; and
- constraints on battery life – the robots have a limited amount of time that they can perform the study before charging and additional time requires the ability to save and load the robot's "state".

Although real-robots does present several challenges to a study, overcoming these challenges is an important part of the solution to robot lexicon learning. The real-world noise motivates methods for categorization. Interfacing robot platforms to the real-world environment requires careful consideration. Finally, the constraints on battery life require planning around poverty of stimulus from two few conversations, and ways to save and load the robots' maps and lexicons so that the study time can be increased.

Adding cognitive and sensory features: New representations were required for each new referent that the L2 robots could learn and link to terms: i) durations – words were linked directly to a number in seconds; ii) cyclic time – words were linked to a pairing of sunlight brightness and the brightness derivative; iii) locations grounded in different cognitive architectures – words were linked differently by different agents, one linking words to nodes in a topological map, and the other linking words to grid squares in an

occupancy grid; and iv) locations and durations with referential uncertainty – words were linked to both locations and durations and KL-divergence was used to select the more informative feature.

L2 was designed to be more extensible than the previous Lingodroids, with interfaces for adding new features (see Section 3.7), which allowed the L2 framework to be easily adapted to learning terms grounded in new features. A key part of this extensibility is a common interface for features. The sufficient interface for the L2 framework is that any pair of features (on a single agent) can produce a value that is considered the *abstract distance* between the features. Just this interface is enough to associate and categorize features.

Adding conversations: New conversations were required to establish different types of associations, as in L2 the grammar of the conversation designated the responses that were permitted. The L2 framework provided common interfaces for conversations, that allowed the greeting and turn-taking to be shared amongst all the learning conversations: *when-did-we-last-meet*, *what-time-of-day-is-it*, *where-in space-time-are-we*, *how-far* and *what-direction*. The interfaces for these conversations included a set of features to be considered and methods to provide the next term to "speak" and handle the last "heard" term. The conversations were reliable enough to create coherent languages after the following measures were taken to prevent them from breaking in specific ways:

- restricting words such as "hello" or "where" so that they could not be used to name features;
- restricting communication when not within shared attention; and
- communicating a "failed" term to reverse the association.

However, although rare, it was still possible for two robots to hold a conversation where they both thought they were speaking and used different names. The spurious conversations introduced some noise into the lexicons, but were only a minor disruption, as anomalies were corrected by further conversations.

A testing language game was added to the L2 framework: the *meet-at* game. The *meet-at* game was inspired from the *goto* game from previous Lingodroid studies (Schulz et al., 2011a), but was used to test spatial and temporal terms. The structure of the *meet-at* game was necessarily different from the learning conversations, as the robots had to stop communicating, move to another location, then resume communicating. To accommodate this, the *meet-at* game had an extra state, ACTING, which was dependent on the robot's location. The trials of the *meet-at* game were only partially successful, with 14/25 successful attempts for the durations study, and 7/10 for the cyclic-time study. The major issue was the robots' ability to navigate within a reasonable time frame. This was partially due to the limited obstacle avoiding sensors on the iRat (three IR sensors) and the need to constantly localize in the map to navigate to goals. In previous Lingodroid studies, where the *goto* game was used with more success, the surrounding ultra-sonic sensors and omni-directional cameras onboard the Pioneer 3-DX robots helped both navigation and localization. Either adding additional obstacle sensors to the iRat, or recording the sequences of left and right wall following, may provide better strategies for the iRat's navigation.

Different cognitive architectures: The use of different cognitive architectures affected the L2 robots' conversations. The *where-are-we* and *where-in-space-time-are-we* conversations had to handle the different cognitive architectures of the L2 robots. The symbols created by the conversations were shared by each

robot but referred to different internal representations. For the *where-are-we* conversation, the L2 robots knew *a priori* that the grammar of the conversation designated their individual spatial representations. For the *where-in-space-time-are-we* conversation, the prior knowledge was relaxed, with the L2 robots having to decide between spatial and temporal words. A key feature of the L2 robots was their ability to automatically link their different underlying spatial representations through word usage and the environment. Results of the studies demonstrated only slightly lower similarity of centroids for the *where-in-space-time-are-we* conversation compared to the *where-are-we* and *when-did-we-last-meet* conversations. The lower similarity appeared to be caused by rare outliers where the robots associated the same word with different dimensions. The studies considered words that could possibly be used by either robot according to the information gain of the word. However, one dynamic of the *meet-at* game that was not captured in the XSL study was the likelihood of a word being chosen for a meeting. Some of the words considered in the centroid comparison would have had very low likelihood of being chosen for a practical *meet-at* game due to competition from other words.

The conversations *how-far* and *what-direction* were used by the L2 framework for grounding transfer across different cognitive architectures. A key characteristic of the L2 framework was that these conversations allowed grounding transfer to develop symbols that are identically grounded although they are based on spatial terms grounded in different cognitive architectures. This is described further below (Section 8.5).

Cross-situational learning: The addition of cross-situational learning to the L2 framework allowed the robots to autonomously link referents to symbols using the information between the use of a symbol and the distance from a feature. The *where-in-space-time-are-we* conversation allowed the collecting of multiple concept-elements for each symbol and then a search for the maximum KL-divergence was used as a weighting for the link between words and meanings. This weighting was used to influence the production, generalization and comprehension of the Lingodroids. KL-divergence was used as the metric instead of mutual information, because where mutual information is the information one *distribution* provides about another, KL-divergence is the information one *event* provides about a distribution. Using mutual information therefore assumes that a word and feature will always be present together ($w \iff f$, for a word w and feature f). KL-divergence instead assumes only that if a word is present, the feature will be present, but not the opposite ($w \rightarrow f$) (see Study IV, Chapter 7).

8.3.2 Limitations of the L2 framework

Providing a framework that is comprehensive across grounding in cognitive processes, grounding in different cognitive architectures and referential uncertainty is a difficult task. The L2 framework has limitations in scalability and the scenarios that are described below.

Grounding limited to space and time: Grounding in cognitive processes has been shown to be crucial for space (Jung and Zelinsky, 2000; Schulz et al., 2011a), time (Studies I and II, Chapters 4 and 5), events (Steels and Baillie, 2003; Schulz et al., 2011b) and actions (Marocco et al., 2010; Tikhanoff et al., 2011), but there are many other cognitive processes used and referred to by humans speaking natural language.

For example, the ability for robots to ground numerical symbols, symbols relating to theory of mind or symbols relating to emotions will all require complex cognitive models. The L2 framework is restricted to the cognitive models that it has access to, which only include spatial and temporal cognition.

No influence of language on cognition: The Sapir-Whorf hypothesis suggests that language can influence cognition (Sapir, 1921; Whorf, 1956) and previous studies have demonstrated how language can be used to correct cognitive maps (Walter et al., 2013). However, a limitation of the L2 framework is that while changes in the robots' cognitive maps can cause changes in the meanings of spatial terms, spatial terms can not cause changes to the robots' cognitive maps. A major challenge of changing the map in response to language is that after a word is used that does not match previous usages of this word, the robot cannot know if the map is wrong, if the other robot is wrong, or if the original definition of the word is wrong. It may be possible to solve this problem by relaxing some of these differences simultaneously.

Limitations on the differences between robots: The studies into different cognitive architectures in this thesis involve robots that are somewhat different, but still have much in common. Highly specialist heterogeneous robots can have architectures that have major differences in cognition, sensors and actuators. For these robots it would be a challenge to share attention to the same features in the environment, to resolve ambiguity for representations where correlations are not obvious and to autonomously perform shared tasks. A limitation of the L2 framework is a reliance on innate identical communication systems between the two robots: sharing the core behaviors of turn-taking, sharing information over the same medium, and associating and generalizing in the same way. It may be possible to create core behaviors that are not innate and are different by specifying an optimization problem where transfer of information is to be optimized and allowing the agents to incrementally evolve towards this goal. However, identical communication systems may be a requirement for agents' to have equal influence on a developed language.

Limits to what can be learned through cross-situational learning: The robots in the studies in this thesis know, *a priori*, what aspects of cognition that they want to share, which simplifies referential resolution. Communication across different cognition may require correlation between complex transforms of cognitive processes, or correlation between intermediate processes of cognition. L2 does not learn words that refer to both a location and duration simultaneously (c.f. Roy (2002b)) or intermediate states of cognitive processes. In the RatSLAM map, for example, it would be possible that the pose cells or visual templates are better understood by another SLAM system than the experiences. To handle these cases, the L2 robots would need to be extended to deal with more features in shared attention, and therefore higher referential uncertainty.

Scalability of cross-situational learning: A key issue for XSL in the literature is that of computational complexity and therefore learning time (Blythe et al., 2010). Ungrounded cross-situational studies suggest that the ratio of meanings presented in a "conversation" to the total number of meanings is the main factor in computational complexity (Smith et al., 2006). However, referential resolution with continuous features has additional complexity in establishing: i) the bounds of categories; ii) the bounds of dimensions; iii) what constitutes sufficient usage; and iv) what constitutes sufficient information.

The L2 framework in its current form is limited to resolving the two dimensions space and time. To handle cases with many dimensions, the L2 framework would require additional attentional mechanisms to reduce the number of possible dimensions that can be named within a conversation.

Shared attention from overhead camera: An overhead camera is used to establish shared attention between the robots in physical studies (Studies I-III, Chapters 4-6) and simulated proximity is used in the XSL study (Study IV, Chapter 7). The limitations of these methods for proximity detection is that they rely on external hardware and would not be suitable for autonomous robots in new environments. There are several options to resolve this limitation and allow the robots to calculate the proximity on board, including hearing distance (Schulz et al., 2011a), which would be suitable for larger environments, fiducial markers (Vogt, 2002), and robot detectors.

8.4 Towards a comprehensive language-learning framework for mobile robots

The L2 framework was developed to address limitations in previous research when applied to mobile robots. Mobile robots have two important properties that impact language learning: i) space and time form the foundations of a mobile robot's cognition; and ii) mobile robots come with a variety of different cognitive architectures. In previous studies, there was no framework for language learning that addressed both these properties of mobile robots. The development of the L2 framework provides insight into the requirements for such a comprehensive framework for mobile robots as outlined in the following sections.

Shared attention: A key challenge for grounding in cognition is that of shared attention – for space, attention can be shared by being in the same place, for time attention can be shared by referring to a former event. The feature to which attention is shared must be either repeatable (toponyms and cyclic time) or generative (durations, distances and directions) – it must be possible to experience the feature again.

Prior processes: Heterogeneous robots need certain prior processes and knowledge for language learning – they need at least the ability to take turns and transfer information. They also need some representations in common that relate to the shared environment, and prior knowledge about the other robot's categorization.

Representation requirements: The studies with different cognitive architectures indicated spatial limitations on how different two cognitive architectures can be and how informative terms must be for coherence to still be high. The success of the robots with different cognitive architectures relied heavily on the robots localizing similarly within their environment (i.e. both robots have Cartesian representations of space with no discontinuities). The success of cross-situational learning relies on the assumption that terms are more informative about their intended dimension than they are about other dimensions over a number of conversations.

Categorization and attention affect referential uncertainty: Referential uncertainty is heavily affected by categorization and attention – categorization affects the resulting distributions of a word's meaning and attention limits the number of features that can be associated with any given word. In the XSL study, when a robot used a word it caused the category that the word named to be expanded, and also the referential uncertainty about the word to be reduced. These dynamics are crucial to the success of XSL as a learning strategy for robots.

For attention, the number of features that can be attended to simultaneously affect referential uncertainty and therefore the learning time of the L2 robots. Robots in the Innate Condition of the XSL study were able to associate a term in one shot due to attention only focused on one feature, but robots in the XS Condition required several presentations.

Links between words and meanings: The links between words and meanings are key to what terms robots can learn and how long it takes them to learn. In each of the temporal studies, the addition of temporal features and the ability to link them to symbols extended the robots' possible terms. In the XSL study, forming many-to-many relationships between words and dimensions extended the robots' learning abilities to unknown dimensions, but increased learning time.

8.5 Impact on robots, grounding and language

This thesis impacts on a variety of topics relating to robots, grounding and language. Many of these topics have been studied in isolation, while the L2 framework integrates the topics together. The impacts on each topic are described in each of the following paragraphs.

Temporal cognition: Previous studies of temporal terms have looked at shared events (Steels and Baillie, 2003), ungrounded temporal ontologies (De Beule, 2006), and grounding events in shared journeys (Schulz et al., 2011b). The L2 framework extends the grounding of durations to practical tasks and adds the ability to ground terms in cyclic events, such as the sunlight levels in a day. Understanding and communicating about time is particularly important for mobile robots for planning motion. The abilities of the L2 robots to use both types of learned temporal terms and spatial language for practical tasks demonstrates important steps towards robots that can autonomously learn the lexicons and communication required for collaborative action.

There are key properties of time itself that affect how it can be grounded and communicated, and these are well described by the limitations of spatial metaphors for time (Galton, 2011). The transience of time is an important consideration for any groundings of time. Time is often represented as a single dimension, but unlike the spatial dimensions, it is not possible to return to a point in time, and therefore grounding symbols in *temponyms* allows a symbol to only refer to past events. This is why previous studies have used discrimination between events (Steels and Baillie, 2003; De Beule, 2006) or the time between events (Schulz et al. (2011b) and Study I, Chapter 4) to learn temporal terms or ideas.

Grounding temporal terms in predictable events, such as sunlight events, provides an alternative to grounding in the differences between events and solves the problems presented by transience, as the grounded event can provide a backdrop against which to synchronize other important events. The L2

robots are the first to robots to learn temporal terms for times of day; however, the Amondawa tribe is a human example of grounding words within cyclic time, as they have words for times of day, but no clocks and no word for "time" (Sinha et al., 2011). The L2 robots are able to learn Amondawa-like temporal concepts, which is an important step towards understanding time in natural language.

Symbol grounding across different cognitive architectures: Previous studies into heterogeneous robots have demonstrated joint tasks with ungrounded communication, or grounded communication with sensory differences but identical cognition (Jung and Zelinsky, 2000). Jung and Zelinsky demonstrated a practical task in which one robot could communicate the location of dust to another robot to decrease the time taken to vacuum an area.

L2 extends the grounding of terms to different sensors and cognition, allowing robots running different underlying SLAM systems to come to agreements on spatial terms. The importance of different cognition is that two robots may need to specialize in different areas that require different representations. In the study of Jung and Zelinsky, the different sensors were enough to share specialist information. However, for complex tasks, cognitive processing of information is required to "construe" sensor readings, and mobile robots will need to be able to share this cognitive insight with other robots and humans. For a cleaning scenario where the two robots have different spatial representations, symbols grounded in different cognition would be required for the robots to communicate about the task.

Cross-situational learning: Previous XSL studies have demonstrated ungrounded mathematical models (Vogt, 2002; Smith et al., 2006; Blythe et al., 2010) or rule based models (Siskind, 1996), where the studies in this thesis demonstrate the importance of considering grounded language when resolving referential uncertainty. While previous studies make interesting observations about the learning times of cross-situational learning (Siskind, 1996; Vogt, 2002; Smith et al., 2006; Blythe et al., 2010), the complex interactions between categorization of real-world sensory data and referential resolution affect all aspects of cross-situational learning. Where in ungrounded frameworks it is possible to count combinations of symbols used together with meanings, for symbols grounded in continuous features the similarity between features needs to be addressed. Features must be compared and organized into categories.

Beyond solving the symbol grounding problem: Even though Steels (2008) claimed that the symbol grounding problem was solved, the studies in this thesis demonstrate that there are many aspects of symbol grounding that were not covered by Steels' solutions. Symbol grounding as defined by Peirce (1974) is extremely versatile, as any part of the semiotic triangle can be abstracted by suitable agents.

The current studies, along with previous Lingodroid studies (Schulz et al., 2011a) demonstrate how symbols can be grounded in the abstraction of an experience, but then as the map is corrected, the experiences can move to a new position, changing the definition of the underlying symbol, but also correcting it to be coherent with other agents. The implementation of robots with different cognitive architectures demonstrated how symbols can be grounded in the environment but are actually dependent on different underlying sensors and algorithms to form the representations required to make the symbol–referent connection (Jung and Zelinsky, 2000). The L2 cross-cognition studies showed how grounding transfer (see Cangelosi et al. (2000)) can be used to develop new groundings by providing groundings in different

cognitive architectures (Studies III and IV, Chapters 6 and 7). The robots use grounding transfer to create identical groundings from their different underlying sensors and cognition. These identical groundings are interesting, because previously the robots had used different sensors and cognition to develop identical *symbols*, but in this study the robots used their identical symbols to develop identical *meanings* (identical private grounding).

Robot-human interactions: These studies have implications for robot-human interactions. In order for robots to understand humans, robots need to be able to understand human spatial and temporal concepts (Schulz et al., 2011a, 2011b). Robots therefore need to be able to ground language in cognition to capture aspects of the spatial and temporal cognition of humans.

Several previous studies have looked at grounding in cognitive processes and referential uncertainty between human and robot (Roy, 2002a; Steels and Kaplan, 2002; Cangelosi et al., 2010; Tellex et al., 2011). These studies have had to compensate for the sensory and cognitive differences between human and robot. Learning between heterogeneous robots reflects these sensory and cognitive differences between humans and robots (see Study III, Chapter 6). The robots with different cognitive architectures in the studies in this thesis require certain shared abilities, such as conversations, comprehension, generalization and social conventions. These abilities suggest beneficial properties for robots to facilitate communication with humans. In particular, robots that can model the comprehension and generalization of humans will be able to determine accurately when to use a term to describe a feature (Oliphant and Batali, 1997). Conversely, the cognitive differences between how humans and robots represent the world is partially addressed by sharing the same environment and reflecting that structure in interactions.

Human language learning: Robot language learning has been used to model the evolution of human languages (Steels, 2015). There are several similarities between the robot studies presented in this thesis and those observed in psychological studies (Akhtar and Montague, 1999; Galantucci, 2005; Smith and Yu, 2008; Sinha et al., 2011). Robots are often used as tools for exploring the origins of language and specific emergent phenomena (Steels, 2006) (and other biological systems, see Webb (2001)). The representations of robots can be analyzed in ways that are unfeasible for humans.

As previously described, the temporal symbols explored in L2 have analogues in humans (Sinha et al., 2011), and the cyclic-time study provides insight into some of the challenges humans might have if learning and planning tasks with only terms for event-based time.

Psychological studies into referential uncertainty have established XSL as biologically plausible for young humans for both objects (Smith et al., 2006; Smith and Yu, 2008) and dimensions (Akhtar and Montague, 1999). The L2 XSL study explores this process and allows analysis of the representations of words and meanings over the course of the study (Study IV, Chapter 7). The results of the L2 XSL study predicts that if one could analyze the representations of a participant in a psychological cross-situational learning study, they would find after one trial the participant had developed multiple, highly-specific sensory meanings associated with a single word. After further presentations the specific meanings would generalize until only one associated meaning was still informative (see Figure 7.7).

One of the areas where the studies in this thesis differ from conventional psychological paradigms is that the Lingodroid studies observe bootstrapping language, while conventional psychological studies

observe the learning of a predefined language (Bloom and Lahey, 1978; Bloom, 2002). An exception is the excellent psychological study of Galantucci (2005) (see also Steels (2006) for a review), which uses a unique methodology to force human participants to bootstrap a symbolic language. Galantucci's study requires participants to play a simple computer game in which the participants cannot see or hear each other, but they can communicate by drawing on a "moving tablet". The simulated motion of the tablet prevents participants from drawing their native symbols (e.g. letters or words from English). The XSL studies (Study IV, Chapter 7) can predict the likely changes between symbols and meanings over time, which gives some insight into the dynamics of language learning. The robot studies also complement Galantucci's studies in several aspects: i) the implementation of a shared experience for assigning a symbol to a place (Galantucci's participants cannot see each other, so they cannot point); ii) the way the subjects generalize their words for places; iii) online learning; and iv) the necessity of invention when there is no word to describe a place. The L2 studies demonstrate sufficiencies (and in some cases inform requirements) for lexicon learning and allow future psychological experimenters to consider environments, communication mediums, games and cognition that are sufficient for language learning.

Robot intelligence: Language and communication is one of the defining features of human intelligence and has long been considered a benchmark for software intelligence (Turing et al., 2004) and robot intelligence (Steels, 1999). Studies into robots that can bootstrap and use language extend the state-of-the-art in robot intelligence. In the Chinese room experiment, Searle suggests that an agent cannot be considered "strong AI" if it is detached from the outside world (Searle, 1980). The studies in this thesis, along with the other symbol grounding frameworks (Steels, 1999; Roy, 2002a; Tellex et al., 2011; Spranger et al., 2014) exhibit capabilities that will be required to approach strong AI.

Subjective experience: In 1974, Nagel wrote an influential article *What is it like to be a bat?* which suggested that inferring consciousness was akin to "being able to be like" (Nagel, 1974). Nagel reasoned that it would never be possible to have the same subjective experiences as another conscious being. While the L2 framework does not address Nagel's arguments on consciousness, robots with different cognitive architectures provide an interesting perspective on reference to subjective experience, as they can learn to communicate about aspects of their environment without having the sensors and representations to be able to "mentally transfer" from one to the other. This leads to the interesting insight that shared biology is not a prerequisite for shared symbols (Study III, Chapter 6).

An integrated framework: L2 integrates spatial and temporal cognition, different cognitive architectures, and referential resolution, into a single framework. The integration of these features reveals dependencies that would otherwise be ignored when solving each problem individually.

In an (in)famous 1973 psychology review entitled *You can't play 20 questions with nature and win*, Newell declared that

> "... Science advances by playing twenty questions with nature. The proper tactic is to frame a general question, hopefully binary, that can be attacked experimentally. Having settled that bits-worth, one can proceed to the next. The policy appears optimal – one never risks much,

> there is feedback from nature at every step, and progress is inevitable. Unfortunately, the questions never seem to be really answered, the strategy does not seem to work." (Newell, 1973)

Newell was referring to the large amount and diversity of the phenomena that was discovered in the field of psychology, and the relative lack of theories that could explain multiple phenomena simultaneously. His analogy was that the theories used to explain phenomena were often single binary oppositions, and that binary questions would never aggregate to explain psychology. Newell's position was subsequently rebutted in another review – *How to win at 20 questions with nature* (Simon, 1980), which asserted that there were extensive theories, and that the binary oppositions were an important part of theory design.

Despite the positions of Newell and Simon perceived as a dichotomy, both Newell and Simon agreed on the general necessity for holistic theoretical modeling to test experimentally observed phenomena. The L2 framework reflects this necessity, as studying "phenomena" in isolation can cause a framework to omit important interplays. An example from the L2 framework is the way in which categorization affects cross-situational learning. While studies that do not implement categorization (e.g. the XSL study of Smith et al. (2006)) are still important, they can not capture all of the dynamics of learning.

8.6 Conclusions

There are three major conclusions of this thesis relating to i) connecting cognitive maps, ii) changes for learning and iii) semiotics.

Connecting cognitive maps: Where previous Lingodroids studies showed how to connect private spatial maps between two agents using lexicons (Schulz et al., 2011a), this thesis has demonstrated the L2 framework is able to i) connect private temporal cognition, ii) connect private, cognitively different spatial maps, and iii) connect private space-time maps with referential uncertainty; while maintaining a stable Lingodroids core.

Changes for learning: The changes that were required to the Lingodroids framework reflect the important aspects of how the particular problems were addressed. The changes included i) new features, ii) new conversations, iii) temporal cognition, iv) methods for cross-situational learning, and v) methods for referential resolution.

Semiotics: This thesis presents an interesting perspective on Peirce's semiotic triangle (Ogden and Richards, 1923). Previous research suggests that there are differences between lexicons for individuals within a population of language learning agents (Steels, 2015), the studies in this thesis suggest that for robots with different cognitive architectures, there are fundamental differences that cannot be corrected by language learning. Where in other studies, learning leads to a convergence of the population towards identical grounded lexicons (Steels, 2015); for the studies in this thesis language learning corrects differences with respect to the environment, but not with respect to individual representations.

8.7 Future work

There are some exciting areas for future work in temporal cognition, different cognitive architectures and referential uncertainty.

Temporal cognition: Temporal cognition has not been researched to the same extent as spatial cognition (Maniadakis and Trahanias, 2011), and as such models of temporal cognition for mobile robots are very limited. For robots to understand all the different uses of time in natural language, more complex temporal cognition is required.

Different cognitive architectures: The robots in this thesis learn shared words for spatial locations (in the environment) that they are both capable of representing. An interesting question is: *how could mobile robots learn shared terms for representations that only one robot has?* This thesis suggests that any term that is informative about a shared dimension can be linked to that dimension through KL-divergence. A word should be considered to be a subjective representation that only one robot has if the word is not informative about any representation. A further question is: *what would it take for a robot to represent these terms that cannot be grounded and use them practically?*

Referential uncertainty: This thesis shows that two robots can bootstrap a small language with 2 dimensions – *what would it take to bootstrap a large language with many dimensions?*

References

Akhtar, N. & Montague, L. (1999). Early lexical acquisition: The role of cross-situational learning. *First Language*, *19*(57), 347–358.

Allen, J. F. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM*, *26*(11), 832–843.

Anderson, J. R., Matessa, M., & Lebiere, C. (1997). ACT-R: A theory of higher level cognition and its relation to visual attention. *Human-Computer Interaction*, *12*(4), 439–462.

Arleo, A. & Gerstner, W. (2000). Spatial cognition and neuro-mimetic navigation: a model of hippocampal place cell activity. *Biological Cybernetics*, *83*(3), 287–299.

Aron, J. (2011). How innovative is Apple's new voice assistant, Siri? *New Scientist*, *212*(2836), 24.

Bailey, T., Nieto, J., Guivant, J., Stevens, M., & Nebot, E. (2006). Consistency of the EKF-SLAM algorithm. In *Proceedings of the IEEE international conference on intelligent robots and systems* (pp. 3562–3568).

Ball, D., Heath, S., Milford, M., Wyeth, G., & Wiles, J. (2010). A navigating rat animat. In *Proceedings of the international conference on the synthesis and simulation of living systems* (pp. 804–811).

Ball, D., Heath, S., Wiles, J., Wyeth, G., Corke, P., & Milford, M. (2013). OpenRatSLAM: An open source brain-based SLAM system. *Autonomous Robots*, *34*(3), 149–176.

Ball, D., Heath, S., Wyeth, G., & Wiles, J. (2010). iRat: Intelligent rat animat technology. In *Proceedings of the Australasian conference on robotics and automation* (pp. 1–3).

Baronchelli, A., Felici, M., Loreto, V., Caglioti, E., & Steels, L. (2006). Sharp transition towards shared vocabularies in multi-agent systems. *Journal of Statistical Mechanics: Theory and Experiment*, *2006*(6), 6–14.

Barrera, A. & Weitzenfeld, A. (2008). Biologically-inspired robot spatial cognition based on rat neurophysiological studies. *Autonomous Robots*, *25*(1-2), 147–169.

Barsalou, L. W. (1999). Perceptual symbol systems. *Behavioral and Brain Sciences*, *22*(04), 637–660.

Bhattacharya, P. & Gavrilova, M. L. (2008). Roadmap-based path planning-using the Voronoi diagram for a clearance-based shortest path. *IEEE Robotics and Automation Automation Magazine*, *15*(2), 58–66.

Billard, A. & Hayes, G. (1997). Learning to communicate through imitation in autonomous robots. In *Proceedings of the international conference on artificial neural networks* (pp. 763–768). Springer.

Bloom, L. & Lahey, M. (1978). *Language development and language disorders*. Wiley.

Bloom, P. (2002). *How children learn the meanings of words*. MIT press.

Blythe, R. A., Smith, K., & Smith, A. D. (2010). Learning times for large lexicons through cross-situational learning. *Cognitive Science*, *34*(4), 620–642.

Boroditsky, L. (2000). Metaphoric structuring: Understanding time through spatial metaphors. *Cognition*, *75*(1), 1–28.

Boroditsky, L. (2001). Does language shape thought?: Mandarin and English speakers' conceptions of time. *Cognitive Psychology*, *43*(1), 1–22.

Brooks, R. A. (1982). *Solving the find-path problem by representing free space as generalized cones*. MIT.

Brooks, R. A. (1990). Elephants don't play chess. *Robotics and Autonomous Systems*, *6*(1), 3–15.

Brooks, R. A. (1991). Intelligence without representation. *Artificial Intelligence*, *47*(1), 139–159.

Cangelosi, A. (2006). The grounding and sharing of symbols. *Pragmatics and Cognition*, *14*(2), 275–285.

Cangelosi, A. (2011). Solutions and open challenges for the symbol grounding problem. *International Journal of Signs and Semiotic Systems*, *1*(1), 49–54.

Cangelosi, A., Coventry, K. R., Rajapakse, R., Joyce, D., Bacon, A., Richards, L., & Newstead, S. N. (2005). Grounding language in perception: A connectionist model of spatial terms and vague quantifiers. In A. Cangelosi, G. Bugmann, & R. Borisyuk (Eds.), *Modeling lanugage, cognition and action: Proceedings of the neural computation and pyschology workshop* (Vol. 16, pp. 47–56).

Cangelosi, A., Greco, A., & Harnad, S. (2000). From robotic toil to symbolic theft: Grounding transfer from entry-level to higher-level categories. *Connection Science*, *12*(2), 143–162.

Cangelosi, A., Metta, G., Sagerer, G., Nolfi, S., Nehaniv, C., Fischer, K., Tani, J., Belpaeme, T., Sandini, G., Nori, F., et al. (2010). Integration of action and language knowledge: A roadmap for developmental robotics. *IEEE Transactions on Autonomous Mental Development*, *2*(3), 167–195.

Cangelosi, A. & Riga, T. (2006). An embodied model for sensorimotor grounding and grounding transfer: Experiments with epigenetic robots. *Cognitive Science*, *30*(4), 673–689.

Censei, A. (2008). An ICP varient using a point-to-line metric. In *Proceedings of the IEEE international conference on robotics and automation*.

Cheeseman, P., Smith, R., & Self, M. (1987). A stochastic map for uncertain spatial relationships. In *Proceedings of the international symposium on robotic research* (pp. 467–474).

Choe, Y., Kwon, J., & Chung, J. R. (2012). Time, consciousness, and mind uploading. *International Journal of Machine Consciousness*, *4*(01), 257–274.

Clark, H. (1973). Cognitive development and the acquisition of language. In T. E. Moore (Ed.), (Chap. Space, time, semantics, and the child, p. 308). Academic Press.

Clarke, E. M. & Emerson, E. A. (1982). *Design and synthesis of synchronization skeletons using branching time temporal logic*. Springer.

Coradeschi, S., Loutfi, A., & Wrede, B. (2013). A short review of symbol grounding in robotic and intelligent systems. *KI-Künstliche Intelligenz*, *27*(2), 129–136.

Cornwall, C., Horiuchi, A., & Lehman, C. (2010). Solar calculation details. [Online]. Available http://www.srrb.noaa.gov/highlights/sunrise/calcdetails.html.

Davison, A. J., Reid, I. D., Molton, N. D., & Stasse, O. (2007). MonoSLAM: Real-time single camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *29*(6), 1052–1067.

De Beule, J. (2006). Simulating the syntax and semantics of linguistic constructions about time. In *Evolutionary epistemology, language and culture* (pp. 407–428). Springer.

Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische mathematik*, *1*(1), 269–271.

Doucet, A., De Freitas, N., Murphy, K., & Russell, S. (2000). Rao-Blackwellised particle filtering for dynamic bayesian networks. In *Proceedings of the conference on uncertainty in artificial intelligence* (pp. 176–183). Morgan Kaufmann Publishers Inc.

Durrant-Whyte, H. & Bailey, T. (2006). Simultaneous localization and mapping: Part I. *IEEE Robotics and Automation Magazine*, *13*(2), 99–110.

Engberg-Pedersen, E. (1999). Cognitive semantics: meaning and cognition. In J. S. Allwood & P. Gärdenfors (Eds.), (Chap. Space and time, pp. 131–152). John Benjamins.

Engelhard, N., Endres, F., Hess, J., Sturm, J., & Burgard, W. (2011). Real-time 3D visual SLAM with a hand-held RGB-D camera. In *Proceedings of the RGB-D workshop on 3D perception in robotics* (Vol. 180).

Evans, N. (2010). *Dying words: Endangered languages and what they have to tell us*. Blackwell.

Ferris, B., Fox, D., & Lawrence, N. D. (2007). WiFi-SLAM using Gaussian process latent variable models. In *Proceedings of the international joint conference on artificial intelligence* (Vol. 7, pp. 2480–2485).

Feynman, R. P. (1948). Space-time approach to non-relativistic quantum mechanics. *Reviews of Modern Physics*, *20*(2), 367.

Fontanari, J. F. & Cangelosi, A. (2011). Cross-situational and supervised learning in the emergence of communication. *Interaction Studies*, *12*(1), 119–133.

Fontanari, J. F. & Perlovsky, L. I. (2008). How language can help discrimination in the neural modelling fields framework. *Neural Networks*, *21*(2), 250–256.

Fontanari, J. F., Tikhanoff, V., Cangelosi, A., Ilin, R., & Perlovsky, L. I. (2009). Cross-situational learning of object–word mapping using neural modeling fields. *Neural Networks*, *22*(5), 579–585.

Fox, D., Burgard, W., Dellaert, F., & Thrun, S. (1999). Monte Carlo localization: Efficient position estimation for mobile robots. *Proceedings of the AAAI Conference on Innovative Applications of Arificial Intelligence*, *1999*, 343–349.

Frank, A. (1992). Qualitative spatial reasoning about distances and directions in geographic space. *Journal of Visual Languages and Computing*, (4), 343–371.

Galantucci, B. (2005). An experimental study of the emergence of human communication systems. *Cognitive Science*, *29*(5), 737–767.

Galton, A. (2011). Time flies but space does not: Limits to the spatialisation of time. *Journal of Pragmatics*, *43*(3), 695–703.

Gentner, D. (2001). Spatial schemas and abstract thought. In M. Gattis (Ed.), *Spatial schemas and abstract thought* (Chap. Spatial metaphors in temporal reasoning, pp. 203–222). A Bradford Book.

Gentner, D., Imai, M., & Boroditsky, L. (2002). As time goes by: Evidence for two systems in processing space –> time metaphors. *Language and Cognitive Processes*, *17*(5), 537–565.

Gibson, J. (1975). Events are perceivable but time is not. In *Proceedings of the conference of the international society for the the study of time* (pp. 295–301). Springer-Verlag. New York.

Gibson, T. T. (A)., Heath, S., Quinn, R. P., Lee, A. H., Arnold, J. T., Sonti, T. S., Whalley, A., Shannon, G. P., Song, B. T., Henderson, J. A., & Wiles, J. (2014). Event-based visual data sets for prediction tasks in

spiking neural networks. In *Proceedings of the IEEE international conference on artifical neural networks* (pp. 635–642).

Gil Jones, E., Browning, B., Dias, M. B., Argall, B., Veloso, M., & Stentz, A. (2006). Dynamically formed heterogeneous robot teams performing tightly-coordinated tasks. In *Proceedings of the IEEE international conference on robotics and automation* (pp. 570–575).

Glover, A. J., Maddern, W. P., Milford, M. J., & Wyeth, G. F. (2010). FAB-MAP + RatSLAM: Appearance-based SLAM for multiple times of day. In *Proceedings of the IEEE international conference on robotics and automation* (pp. 3507–3512).

Grisetti, G., Stachniss, C., & Burgard, W. (2007). Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE Transactions on Robotics*, *23*(1), 34–46.

Harnad, S. (1990). The symbol grounding problem. *Physica D: Nonlinear Phenomena*, *42*(1), 335–346.

Hart, P. E., Nilsson, N. J., & Raphael, B. (1968). A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, *4*(2), 100–107.

Heath, S., Ball, D., Schulz, R., & Wiles, J. (2013). Communication between Lingodroids with different cognitive capabilities. In *Proceedings of the IEEE international conference on robotics and automation* (pp. 490–495).

Heath, S., Cummings, A., Wiles, J., & Ball, D. (2011). A rat in the browser. In *Proceedings of the Australasian conference on robotics and automation*.

Heath, S., Schulz, R., Ball, D., & Wiles, J. (2012a). Lingodroids: learning terms for time. In *Proceedings of the IEEE international conference on robotics and automation* (pp. 1862–1867).

Heath, S., Schulz, R., Ball, D., & Wiles, J. (2012b). Long summer days: Grounded learning of words for the uneven cycles of real world events. *IEEE Transactions on Autonomous Mental Development*, *4*(3), 192–203.

Ho, T. S., Fai, Y. C., & Ming, E. S. L. (2015). Simultaneous localization and mapping survey based on filtering techniques. In *In proceedings of the asian control conference* (pp. 1–6).

Hobbs, J. R. & Pan, F. (2004). An ontology of time for the semantic web. *ACM Transactions on Asian Language Information Processing*, *3*(1), 66–85.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*(5), 359–366.

Huang, G. P., Mourikis, A., Roumeliotis, S., et al. (2009). On the complexity and consistency of UKF-based SLAM. In *Proceedings of the ieee international conference on robotics and automation* (pp. 4401–4408).

Hutchins, E. & Hazlehurst, B. (1995). How to invent a lexicon: The development of shared symbols in interaction. *Artificial Societies: The Computer Simulation of Social Life*, 157–189.

Jung, D. & Zelinsky, A. (2000). Grounded symbolic communication between heterogeneous cooperating robots. *Autonomous Robots*, *8*(3), 269–292.

Kambhampati, S. & Davis, L. S. (1986). Multiresolution path planning for mobile robots. *IEEE Journal of Robotics and Automation*, *2*(3), 135–145.

Kant, I. (1998). *Critique of pure reason* (P. Guyer & A. W. Wood, Eds.). Cambridge University Press.

Khaleghi, B., Khamis, A., Karray, F. O., & Razavi, S. N. (2013). Multisensor data fusion: A review of the state-of-the-art. *Information Fusion*, *14*(1), 28–44.

Kirby, S. (2002). Natural language from artificial life. *Artificial life*, *8*(2), 185–215.

Kirby, S. & Hurford, J. R. (2002). The emergence of linguistic structure: An overview of the iterated learning model. In *Simulating the evolution of language* (pp. 121–147). Springer.

Kohlbrecher, S., Von Stryk, O., Meyer, J., & Klingauf, U. (2011). A flexible and scalable SLAM system with full 3D motion estimation. In *Ieee international symposium on safety, security, and rescue robotics* (pp. 155–160).

Kollar, T., Tellex, S., Roy, D., & Roy, N. (2010). Toward understanding natural language directions. In *Proceedings of the ACM/IEEE international conference on human-robot interaction* (pp. 259–266).

Kramer, M. A. (1991). Nonlinear principal component analysis using autoassociative neural networks. *American Institute of Chemical Engineers Journal*, *37*(2), 233–243.

Kuipers, B. (2008). An intellectual history of the spatial semantic hierarchy. *Robotics and cognitive approaches to spatial mapping*, 243–264.

Laird, J. E., Newell, A., & Rosenbloom, P. S. (1987). SOAR: An architecture for general intelligence. *Artificial Intelligence*, *33*(1), 1–64.

Lakoff, G. & Johnson, M. (1980). *Metaphors we live by*. Chicago London.

Latombe, J.-C. (2012). *Robot motion planning*. Springer.

Leonard, J. J. & Durrant-Whyte, H. F. (1991). Simultaneous map building and localization for an autonomous mobile robot. In *Proceedings of the ieee international workshop on intelligence for mechanical systems* (pp. 1442–1447). Ieee.

Levinson, S. C. (1996). Language and space. *Annual Review of Anthropology*, 353–382.

Levinson, S. C. (2003). *Space in language and cognition: Explorations in cognitive diversity*. Cambridge University Press.

Levit, M. & Roy, D. (2007). Interpretation of spatial language in a map navigation task. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, *37*(3), 667–679.

Lichtsteiner, P., Posch, C., & Delbruck, T. (2008). A 128 $\times$ 128 120dB 15μs latency asynchronous temporal contrast vision sensor. *IEEE Journal of Solid-State Circuits*, *43*(2), 566–576.

Maddern, W., Milford, M., & Wyeth, G. (2012). CAT-SLAM: Probabilistic localisation and mapping using a continuous appearance-based trajectory. *The International Journal of Robotics Research*, *31*(4), 429–451.

Mangin, O. & Oudeyer, P.-Y. (2013). Learning semantic components from subsymbolic multimodal perception. In *Proceedings of the IEEE joint international conference on development and learning and epigenetic robotics* (pp. 1–7).

Maniadakis, M. & Trahanias, P. (2011). Temporal cognition: A key ingredient of intelligent systems. *Frontiers in Neurorobotics*, *5*.

Maniadakis, M. & Trahanias, P. (2014). Time models and cognitive processes: A review. *Frontiers in Neurorobotics*, *8*.

Marocco, D., Cangelosi, A., Fischer, K., & Belpaeme, T. (2010). Grounding action words in the sensorimotor interaction with the world: Experiments with a simulated iCub humanoid robot. *Frontiers in Neurorobotics*, *4*.

Meeus, J. (1991). *Astronomical algorithms*. Willmann-Bell, Incorporated.

Metta, G., Sandini, G., Vernon, D., Natale, L., & Nori, F. (2008). The iCub humanoid robot: An open platform for research in embodied cognition. In *Proceedings of the acm workshop on performance metrics for intelligent systems* (pp. 50–56).

Meyer, J.-A. & Filliat, D. (2003). Map-based navigation in mobile robots: II. A review of map-learning and path-planning strategies. *Cognitive Systems Research*, *4*(4), 283–317.

Milford, M. J., Wiles, J., & Wyeth, G. F. (2010). Solving navigational uncertainty using grid cells on robots. *PLoS Computational Biology*, *6*(11).

Milford, M. J. & Wyeth, G. F. (2008). Mapping a suburb with a single camera using a biologically inspired SLAM system. *IEEE Transactions on Robotics*, *24*(5), 1038–1053.

Milford, M. J. & Wyeth, G. F. (2012). SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights. In *Proceedings of the ieee international conference on robotics and automation* (pp. 1643–1649). IEEE.

Milford, M. & Wyeth, G. (2010). Persistent navigation and mapping using a biologically inspired SLAM system. *The International Journal of Robotics Research*, *29*(9), 1131–1153.

Montemerlo, M. & Thrun, S. (2004). A multi-resolution pyramid for outdoor robot terrain perception. In *Proceedings of the AAAI national conference on artificial intelligence* (Vol. 4, pp. 464–469).

Montemerlo, M. & Thrun, S. (2007a). FastSLAM: a scalable method for the simultaneous localization and mapping problem in robotics. (Chap. FastSLAM 2.0, pp. 63–90). Springer.

Montemerlo, M. & Thrun, S. (2007b). *The SLAM problem*. Springer.

Montemerlo, M., Thrun, S., Koller, D., Wegbreit, B., et al. (2002). Fastslam: a factored solution to the simultaneous localization and mapping problem. In *Proceedings of the AAAI conference on innovative applications of articial intelligence* (pp. 593–598).

Moore, K. (2006). Space-to-time mappings and temporal concepts. *Cognitive Linguistics*, (2), 199.

Munguía, R. & Grau, A. (2008). Single sound source SLAM. In *Proceedings of the Ibero-American congress on pattern recognition: Progress in pattern recognition, image analysis and applications* (pp. 70–77). Springer.

Murphy, R. R., Casper, J., Micire, M., Hyams, J., et al. (2000). Mixed-initiative control of multiple heterogeneous robots for urban search and rescue.

Nagel, T. (1974). What is it like to be a bat? *The philosophical review*, *83*(4), 435–450.

Newell, A. (1973). You can't play 20 questions with nature and win: projective comments on the papers of this symposium. In W. G. Chase (Ed.), *Visual information processing*. Academic Press.

Newell, A. & Simon, H. A. (1976). Computer science as empirical inquiry: Symbols and search. *Communications of the ACM*, *19*(3), 113–126.

Núñez, R. E. & Sweetser, E. (2006). With the future behind them: Convergent evidence from Aymara language and gesture in the crosslinguistic comparison of spatial construals of time. *Cognitive Science*, *30*(3), 401–450.

Oates, T. (2003). Grounding word meanings in sensor data: Dealing with referential uncertainty. In *Proceedings of the HLT-NAACL workshop on learning word meaning from non-linguistic data* (pp. 62–69). Association for Computational Linguistics.

Ogden, C. K. & Richards, I. A. (1923). *The meaning of meaning.* Harcourt, Brace.

O'Keefe, J. & Nadel, L. (1978). *The hippocampus as a cognitive map*. Clarendon Press Oxford.

Oliphant, M. & Batali, J. (1997). Learning and the emergence of coordinated communication. *Center for Research on Language Newsletter*, *11*(1), 1–46.

Parker, L. E., Kannan, B., Tang, F., & Bailey, M. (2004). Tightly-coupled navigation assistance in heterogeneous multi-robot teams. In *Proceedings of the IEEE international conference on intelligent robots and systems* (Vol. 1, pp. 1016–1022).

Peirce, C. S. (1974). *Collected papers of Charles Sanders Peirce*. Harvard University Press.

Perlovsky, L. I. (2001). *Neural networks and intellect: Using model-based concepts*. Oxford University Press New York.

Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., & Ng, A. Y. (2009). ROS: An open-source robot operating system. In *Proceedings of the workshop on open source software* (Vol. 3, *3*). 2.

Quine, W. (1960). *Word and object*. Cambridge Technology Press.

Riga, T., Cangelosi, A., & Greco, A. (2004). Symbol grounding transfer with hybrid self-organizing/supervised neural networks. In *Proceedings of the IEEE international joint conference on neural networks* (Vol. 4, pp. 2865–2869).

Roy, D. (2002b). Learning visually grounded words and syntax of natural spoken language. *Evolution of Communication*, *4*(1), 33–56.

Roy, D. (2005). Semiotic schemas: a framework for grounding language in action and perception. *Artificial Intelligence*, *167*(1), 170–205.

Roy, D. K. (2002a). Learning visually grounded words and syntax for a scene description task. *Computer Speech and Language*, *16*(3), 353–385.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1988). Learning representations by back-propagating errors. *Cognitive Modeling*.

Sapir, E. (1921). *Language. an introduction to the study of speech*. Harcourt, Brace and World.

Sariff, N. & Buniyamin, N. (2006). An overview of autonomous mobile robot path planning algorithms. In *Proceedings of the IEEE student conference on research and development* (pp. 183–188).

Schoppers, M. (1987). Universal plans for reactive robots in unpredictable environments. In *Proceedings of the international joint conference on artificial intelligence* (Vol. 87, pp. 1039–1046).

Schulz, R. (2008). *Spatial language for mobile robots: the formation and generative grounding of toponyms* (Doctoral dissertation, University of Queensland).

Schulz, R., Glover, A., Milford, M. J., Wyeth, G., & Wiles, J. (2011). Lingodroids: Studies in spatial cognition and language. In *Proceedings of the IEEE international conference on robotics and automation* (pp. 178–183).

Schulz, R., Heath, S., Gibson, T. T. (A)., & Wiles, J. (2014). Lingodroids - The University of Queensland. [Online]. Available http://www.lingodroids.org.

Schulz, R., Whittington, M., & Wiles, J. (2012). Language change in socially structured populations. In *Proceedings of the international conference on the evolution of language* (pp. 312–319).

Schulz, R., Wyeth, G., & Wiles, J. (2011a). Lingodroids: Socially grounding place names in privately grounded cognitive maps. *Adaptive Behavior*, *19*(6), 409–424.

Schulz, R., Wyeth, G., & Wiles, J. (2011b). Are we there yet? Grounding temporal concepts in shared journeys. *IEEE Transactions on Autonomous Mental Development*, *3*(2), 163–175.

Schulz, R., Wyeth, G., & Wiles, J. (2012). Beyond here-and-now: Extending shared physical experiences to shared conceptual experiences. *Adaptive Behavior*, *20*(5), 360–387.

Searle, J. R. (1980). Minds, brains, and programs. *Behavioral and Brain Sciences*, *3*(03), 417–424.

Simmons, R., Apfelbaum, D., Burgard, W., Fox, D., Moors, M., Thrun, S., & Younes, H. (2000). Coordination for multi-robot exploration and mapping. In *Proceedings of the AAAI conference on innovative applications of articial intelligence* (pp. 852–858).

Simmons, R., Apfelbaum, D., Fox, D., Goldman, R. P., Haigh, K. Z., Musliner, D. J., Pelican, M., & Thrun, S. (2000). Coordinated deployment of multiple, heterogeneous robots. In *Proceedings of the IEEE international conference on intelligent robots and systems* (Vol. 3, pp. 2254–2260). IEEE.

Simon, H. A. (1990). Invariants of human behavior. *Annual review of psychology*, *41*(1), 1–20.

Simon, H. A. (1980). How to win at twenty questions with nature. In R. A. Cole (Ed.), *Perception and production of fluent speech* (Chap. 17, pp. 535–548). Lawrence Erlbaum Associates, Inc.

Sinha, C., da Silva Sinha, V., Zinken, J., & Sampaio, W. (2011). When time is not space: The social and linguistic construction of time intervals and temporal event relations in an Amazonian culture. *Language and Cognition*, *3*(1), 137–169.

Siskind, J. M. (1996). A computational study of cross-situational techniques for learning word-to-meaning mappings. *Cognition*, *61*(1), 39–91.

Sloman, A. & Chappell, J. (2005). The altricial-precocial spectrum for robots. In *Proceedings of the international joint conference on artificial intelligence* (Vol. 19, p. 1187).

Smith, K., Smith, A. D., Blythe, R. A., & Vogt, P. (2006). Cross-situational learning: A mathematical approach. In *Symbol grounding and beyond* (pp. 31–44). Springer.

Smith, L. & Yu, C. (2008). Infants rapidly learn word-referent mappings via cross-situational statistics. *Cognition*, *106*(3), 1558–1568.

Smith, M., Baldwin, I., Churchill, W., Paul, R., & Newman, P. (2009). The new college vision and laser data set. *The International Journal of Robotics Research*, *28*(5), 595–599.

Spranger, M. (2012). The co-evolution of basic spatial terms and categories. *Experiments in cultural language evolution*, 111–141.

Spranger, M. (2013). Evolutionary explanations for spatial language – A case study on landmarks. In *Advances in artificial life* (Vol. 12, pp. 1999–1205).

Spranger, M., Suchan, J., Bhatt, M., & Eppe, M. (2014). Grounding dynamic spatial relations for embodied (robot) interaction. In *Proceedings of the Pacific Rim international conference on aritificial intelligence* (pp. 958–971).

Stachniss, C., Grisetti, G., Hähnel, D., & Burgard, W. (2004). Improved Rao-Blackwellized mapping by adaptive sampling and active loop-closure. In *Proceedings of the workshop on self-organization of adaptive behavior* (pp. 1–15).

Steels, L. (1995). A self-organizing spatial vocabulary. *Artificial Life*, *2*(3), 319–332.

Steels, L. (1999). *The talking heads experiment*.

Steels, L. (2000). The emergence of grammar in communicating autonomous robotic agents. In *Proceedings of the European conference on artificial intelligence* (pp. 764–769).

Steels, L. (2001). Language games for autonomous robots. *IEEE Intelligent Systems*, *16*(5), 16–22.

Steels, L. (2003). Evolving grounded communication for robots. *Trends in Cognitive Sciences*, *7*(7), 308–312.

Steels, L. (2005). The emergence and evolution of linguistic structure: From lexical to grammatical communication systems. *Connection Science*, *17*(3-4), 213–230.

Steels, L. (2006). Experiments on the emergence of human communication. *Trends in Cognitive Sciences*, *10*(8), 347–349.

Steels, L. (2008). The symbol grounding problem has been solved. so what's next. *Symbols and Embodiment: Debates on Meaning and Cognition*, 223–244.

Steels, L. (2015). *The talking heads experiment: origins of words and meanings* (L. Steels & R. van Trijp, Eds.). Computational Models of Language Evolution. Habelschwerdter Allee 45 14195 Berlin, Germany: Language Science Press. Retrieved from http://langsci-press.org/catalog/book/49

Steels, L. & Baillie, J.-C. (2003). Shared grounding of event descriptions by autonomous robots. *Robotics and Autonomous Systems*, *43*(2), 163–173.

Steels, L. & Belpaeme, T. (2005). Coordinating perceptually grounded categories through language: A case study for colour. *Behavioral and Brain Sciences*, *28*(4), 469–488.

Steels, L. & Kaplan, F. (2002). Aibo's first words: The social learning of language and meaning. *Evolution of Communication*, *4*(1), 3–32.

Steels, L., Loetzsch, M., & Spranger, M. (2007). Semiotic dynamics solves the symbol grounding problem. *Nature Preceedings*, *1*.

Steels, L. & Vogt, P. (1997). Grounding adaptive language games in robotic agents. In *Proceedings of the European conference on artificial life*. Cambridge, MA.

Stramandinoli, F., Cangelosi, A., & Marocco, D. (2011). Towards the grounding of abstract words: A neural network model for cognitive robots. In *Proceedings of the international joint conference on neural networks* (pp. 467–474).

Stramandinoli, F., Marocco, D., & Cangelosi, A. (2012). The grounding of higher order concepts in action and language: A cognitive robotics model. *Neural Networks*, *32*, 165–173.

Sun, R. (2000). Symbol grounding: a new look at an old idea. *Philosophical Psychology*, *13*(2), 149–172.

Taatgen, N. A., Van Rijn, H., & Anderson, J. (2007). An integrated theory of prospective time interval estimation: The role of cognition, attention, and learning. *Psychological Review*, *114*(3), 577.

Tellex, S., Kollar, T., Dickerson, S., Walter, M. R., Banerjee, A. G., Teller, S., & Roy, N. (2011). Approaching the symbol grounding problem with probabilistic graphical models. *AI Magazine*, *32*(4), 64–76.

Thrun, S. (2003). Learning occupancy grid maps with forward sensor models. *Autonomous Robots*, *15*(2), 111–127.

Tikhanoff, V., Cangelosi, A., Fitzpatrick, P., Metta, G., Natale, L., & Nori, F. (2008). An open-source simulator for cognitive robotics research: The prototype of the iCub humanoid robot simulator. In *Proceedings of the workshop on performance metrics for intelligent systems* (pp. 57–61).

Tikhanoff, V., Cangelosi, A., & Metta, G. (2011). Integration of speech and action in humanoid robots: iCub simulation experiments. *IEEE Transactions on Autonomous Mental Development*, *3*(1), 17–29.

Tulving, E. (1983). *Elements of episodic memory*. Clarendon Press.

Turing, A., Braithwaite, R., Jefferson, G., & Newman, M. (2004). Can automatic calculating machines be said to think? In B. J. Copeland (Ed.), (Chap. 14, p. 487). Oxford University Press.

Uno, R., Marocco, D., Nolfi, S., & Ikegami, T. (2011). Emergence of protosentences in artificial communicating systems. *IEEE Transactions on Autonomous Mental Development*, *3*(2), 146–153.

Varzi, A. C. (2007). Handbook of spatial logics. (Chap. Spatial reasoning and ontology: Parts, wholes, and locations, pp. 945–1038). Springer.

Vaughan, R. (2008). Massively multi-robot simulation in Stage. *Swarm Intelligence*, *2*(2-4), 189–208.

Vera, A. H. & Simon, H. A. (1993). Situated action: A symbolic interpretation. *Cognitive Science*, *17*(1), 7–48.

Vogt, P. (2002). The physical symbol grounding problem. *Cognitive Systems Research*, *3*(3), 429–457.

Vogt, P. (2012). Exploring the robustness of cross-situational learning under Zipfian distributions. *Cognitive Science*, *36*(4), 726–739.

Walter, M. R., Hemachandra, S., Homberg, B., Tellex, S., & Teller, S. (2013). Learning semantic maps from natural language descriptions. In *Proceedings of robotics: science and systems*.

Webb, B. (2001). Can robots make good models of biological behaviour? *Behavioral and brain sciences*, *24*(06), 1033–1050.

Whorf, B. L. (1956). *Language, thought, and reality: Selected writings of Benjamin Lee Whorf* (J. B. Carroll, Ed.). Technology Press.

Wiles, J., Ball, D., Heath, S., Nolan, C., & Stratton, P. (2010). Spike-time robotics: A rapid response circuit for a robot that seeks temporally varying stimuli. In *Proceedings of the international conference on neural information processing*.

Wiles, J., Heath, S., Ball, D., Quinn, L., & Chiba, A. (2012). Rat meets irat. In *Proceedings of the IEEE international conference on development and learning and epigenetic robotics*.

Wittgenstein, L., Anscombe, G. E. M., & Cumming, M. (1958). *Philosophical investigations*. Blackwell.

Yu, C. & Smith, L. B. (2007). Rapid word learning under uncertainty via cross-situational statistics. *Psychological Science*, *18*(5), 414–420.

Zhou, L. & Hripcsak, G. (2007). Temporal reasoning with medical data – A review with emphasis on medical natural language processing. *Journal of Biomedical Informatics*, *40*(2), 183–202.

Ziemke, T. (1999). Rethinking grounding. In A. Riegler & M. Peschl (Eds.), *Does representation need reality? - proceedings of the international conference 'new trends in cognitive science' - perspectives from cognitive science* (pp. 177–190).

Ziemke, T. (2003). What's that thing called embodiment? In *Proceedings of the annual meeting of the cognitive science society* (pp. 1305–1310).

Appendix A

OpenRatSLAM: An open source brain-based SLAM system

This thesis includes the following paper as an appendix:

- Ball, D., Heath, S., Wiles, J., Wyeth, G., Corke, P. and Milford, M. (2013). OpenRatSLAM: An open source brain-based SLAM system. In Pantofaru, C., Chitta, S., Gerkey, B., Rusu, R., Smart, W. D. and Vaughan, R. (Eds.). *Autonomous Robots*, 34(3):149-176.

Only the abstract of the paper is reproduced here. The complete paper is available online.

DOI – 10.1007/s10514-012-9317-9

URL – http://link.springer.com/article/10.1007/s10514-012-9317-9

Abstract – RatSLAM is a navigation system based on the neural processes underlying navigation in the rodent brain, capable of operating with low resolution monocular image data. Seminal experiments using RatSLAM include mapping an entire suburb with a web camera and a long term robot delivery trial. This paper describes OpenRatSLAM, an open-source version of RatSLAM with bindings to the Robot Operating System framework to leverage advantages such as robot and sensor abstraction, networking, data playback, and visualization. OpenRatSLAM comprises connected ROS nodes to represent RatSLAM's pose cells, experience map, and local view cells, as well as a fourth node that provides visual odometry estimates. The nodes are described with reference to the RatSLAM model and salient details of the ROS implementation such as topics, messages, parameters, class diagrams, sequence diagrams, and parameter tuning strategies. The performance of the system is demonstrated on three publicly available open-source datasets.

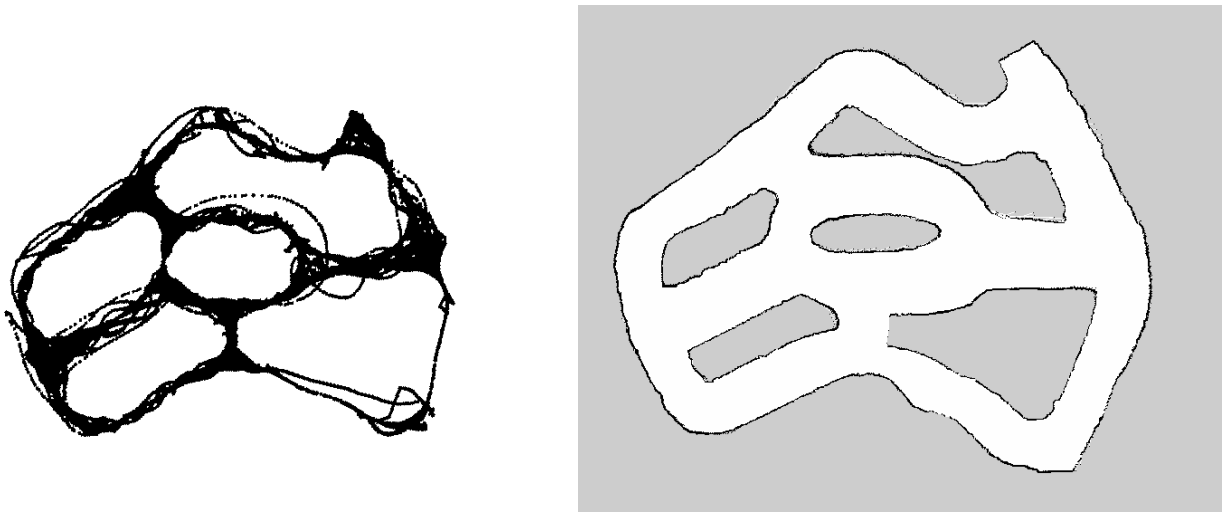
Appendix B

Robot maps, lexicons and learning dynamics

This section contains supplementary material for Study IV (Chapter 7) that was omitted from the submitted paper due to page limits for the journal; however, the following figures were taken from the same datasets used for generating the results in the XSL studies. The figures in this section are provided to give further visual insight into the results of this chapter. The figures visualize components of the L2 framework: the maps and lexicons the robots produced, the differences between two robots' lexicons, an extended version of the *select words* figure (Figure 7.7) and further information phase portraits for comparison.

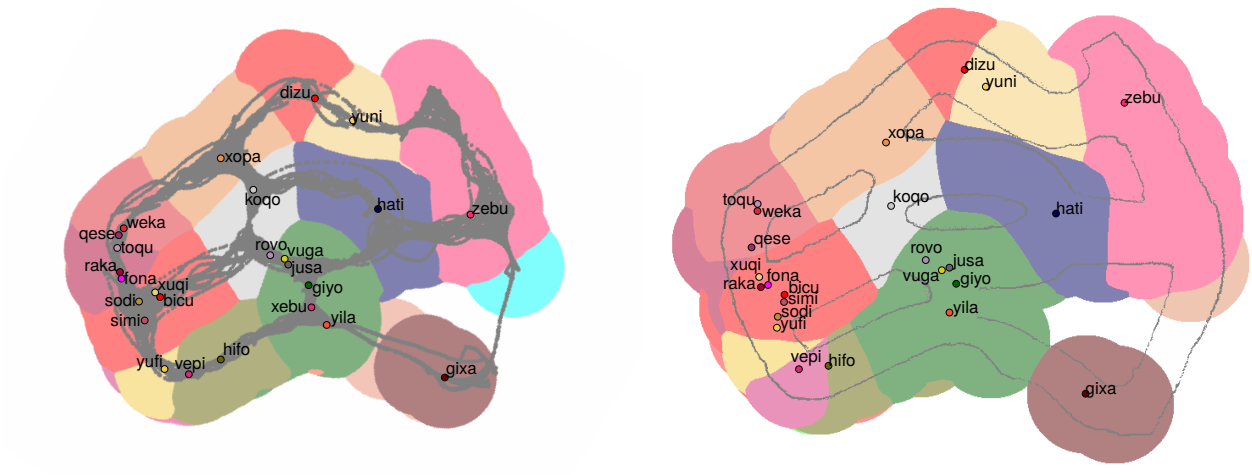
Spatial maps and lexicons: The Lingodroids within the Stage simulator developed maps with their different sensors and cognition. Extended versions of the RatSLAM and Gmapping maps are shown in Figure B.1. The maps were used as the grounding for spatial lexicons. The spatial lexicons can be imposed onto the robots maps (Figure B.2). Temporal lexicons were learned using the robots' clocks as temporal cognition. The temporal lexicons cover all durations from 0-550s (see Figure B.3). The lexicons were compared through the centroids (or exemplars) of each term. The centroids were transformed from the map back to the simulated environment. The distances between spatial and temporal centroids were then plotted (see Figures B.4).

Words, meanings and information phase portraits: For the publication of Study IV (Chapter 7), the figure of words, their spatial meaning, temporal meaning and information phase portraits was truncated due to space restrictions. An extended figure with eight words is shown in Figure B.5. A larger set of information phase portraits were created for the same lexicons. These are created by following the information gain about space and the information gain about time across all of the conversations in the study. Six additional words (in a larger size) are shown below (Figure B.6). The word *bicu* "spikes" with a large magnitude, indicating that it was used to describe a rare duration during one of the presentations, but subsequent conversations have re-established it as a spatial term. The other terms have the same general pattern as the previous plots: the listener starts with a high information gain about space and time, and reduces one dimension more than the other.



(a) (b)

Figure B.1: The spatial maps developed by the Lingodroids. a) the topological RatSLAM experience map produced by the iRat; b) the Gmapping occupancy grid produced by the laserbot.



(a) (b)

Figure B.2: Spatial lexicons for the Lingodroids imposed on their underlying maps. a) the iRat's lexicon, b) the laserbot's lexicon.

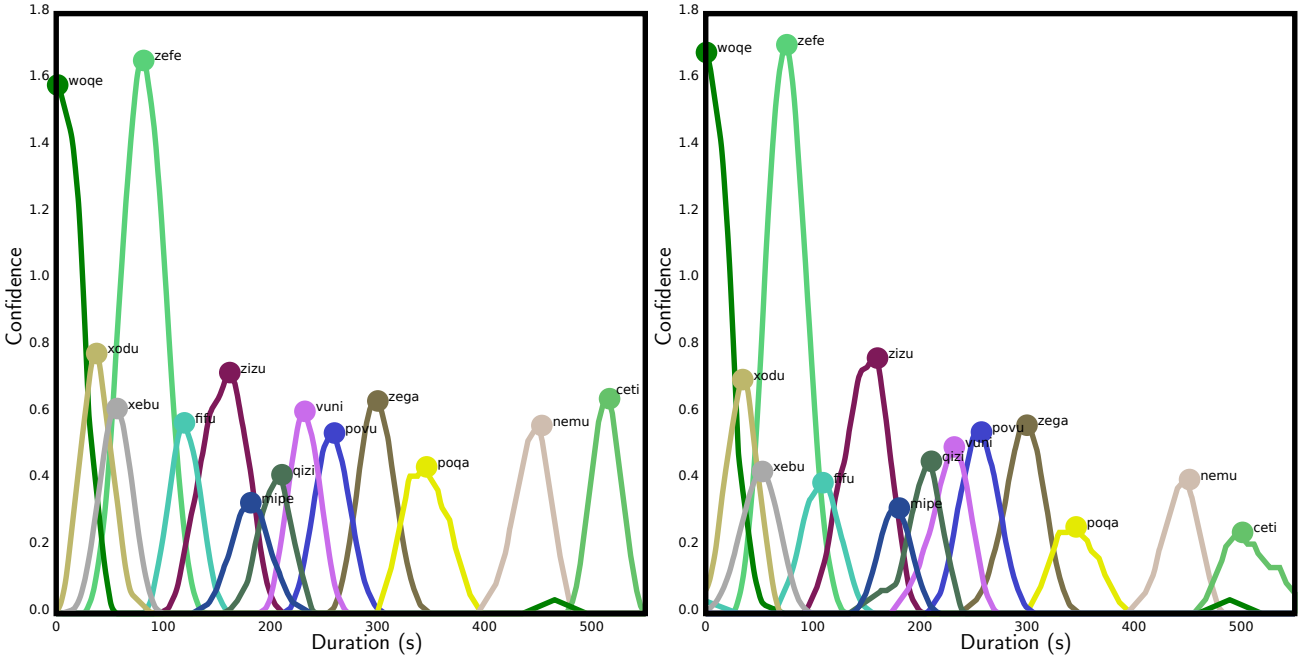


Figure B.3: Temporal lexicons for the two Lingodroids. a) the iRat's temporal lexicon; b) the laserbot's temporal lexicon. Each different colored curve represents a different (labeled) word. The circle imposed on each curve represents the exemplar: the duration that the word best describes.

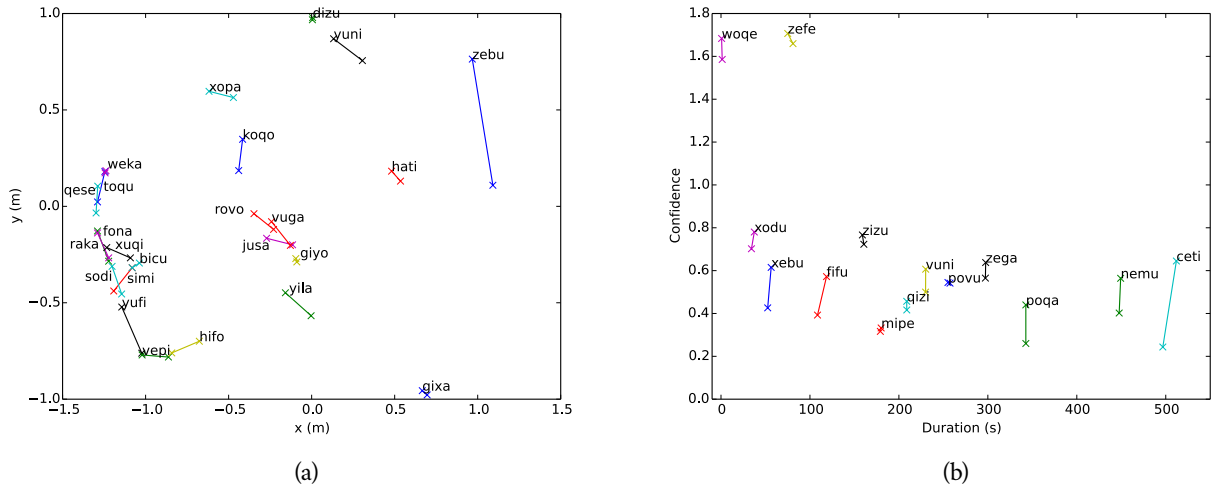


Figure B.4: Spatial and temporal differences between the two Lingodroids' lexicons. a) the spatial differences in meters; b) the temporal differences in seconds on the x axis and confidence on the y axis.

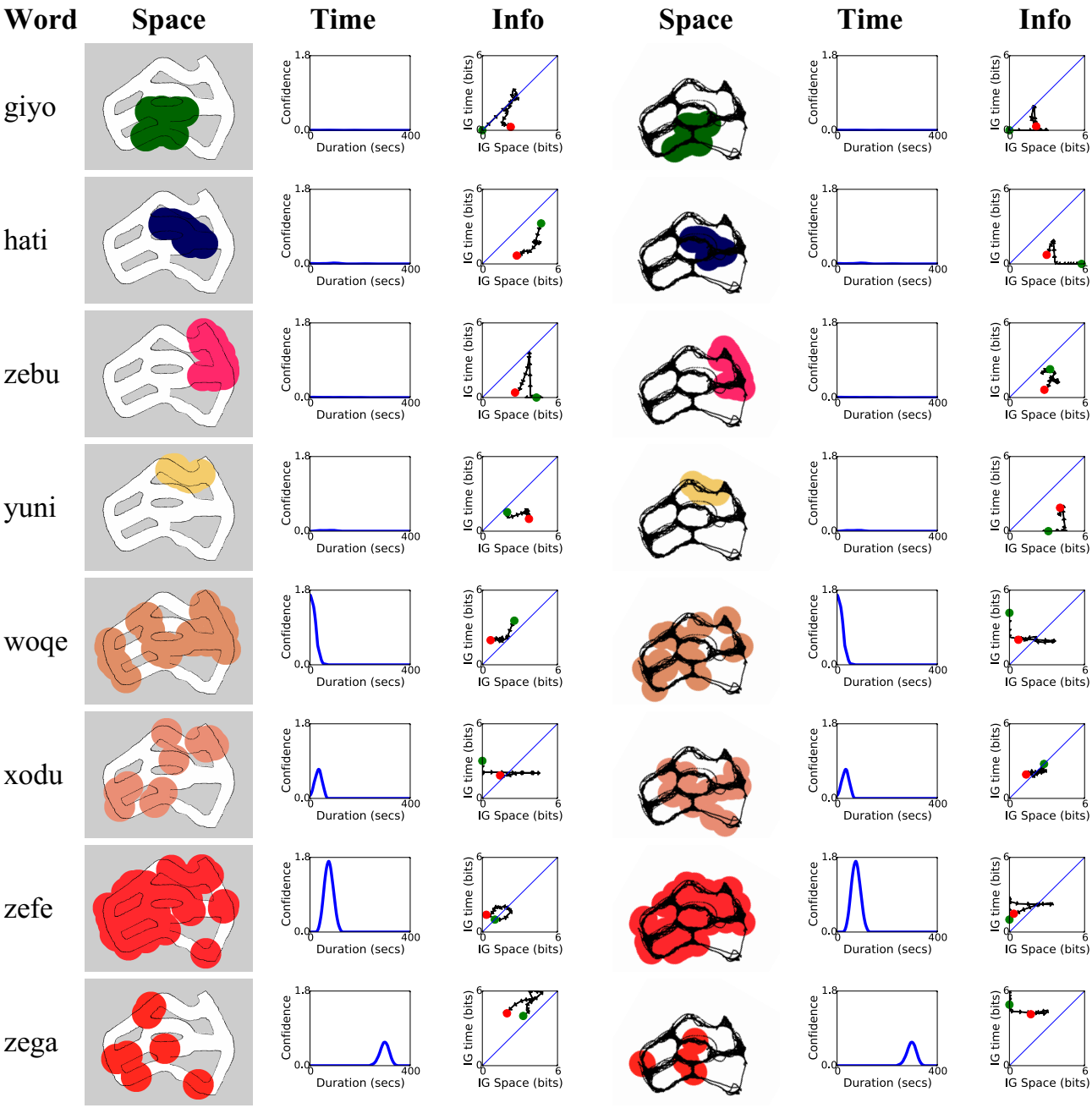


Figure B.5: Words' spatial specification, temporal specification and information journeys for both robots.

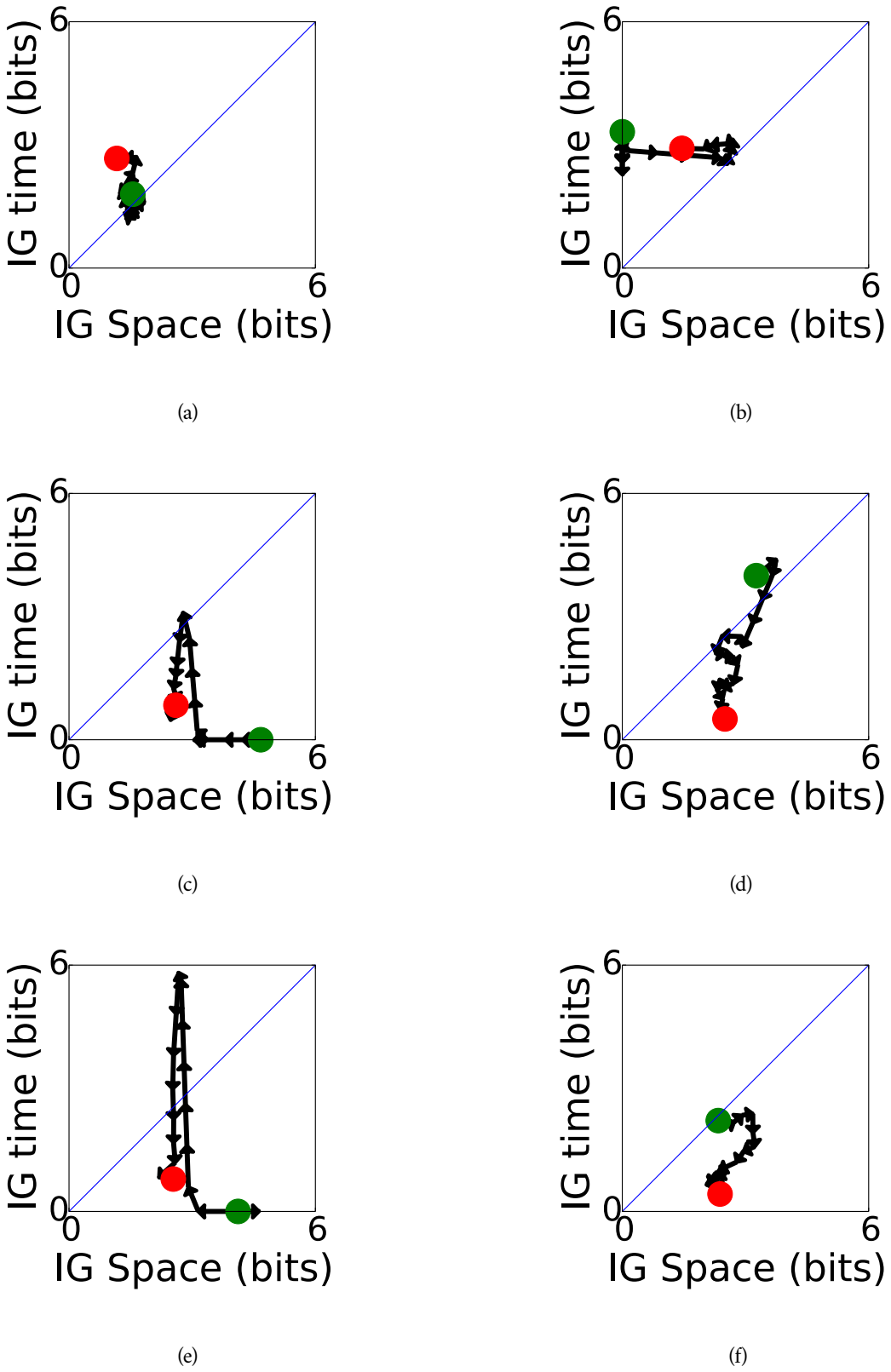


Figure B.6: Information phase portraits for three words. a) and b) zizu for iRat and laserbot; c) and d) xopa for iRat and laserbot; and e) and f) bicu for iRat and laserbot.