The Pennsylvania State University

The Graduate School

College of Information Sciences and Technology

A COLLABORATIVE DIALOGUE APPROACH FOR HUMAN-GIS COMMUNICATION OF VAGUE SPATIAL CONCEPTS

A Thesis in

Information Sciences and Technology

by

Hongmei Wang

Submitted in Partial Fulfillment
of the Requirements
for the Degree of
Doctor of Philosophy

May 2007

The thesis of Hongmei Wang was reviewed and approved* by the following:

Guoray Cai
Associate Professor of Information Sciences and Technology
Thesis Advisor
Chair of Committee

Alan M. MacEachren
Professor of Geography

Michael McNeese
Associate Professor of Information Sciences and Technology

John Yen
University Professor of Information Sciences and Technology

Joseph Lambert
Senior Associate Dean
Head of College of Information Sciences and Technology

*Signatures are on file in the Graduate School

ABSTRACT

Natural multimodal interfaces have been proposed as an alternative interface for geospatial information systems. A fundamental challenge in developing a usable conversational interface for GIS is effective communication of spatial concepts in natural language, which are commonly vague in meaning. This study recognizes and makes distinctions between two sources of vagueness in human-GIS communication: (1) there are multiple contexts within which a spatial concept can be interpreted (i.e. context-dependency); and (2) there are multiple interpretations of the same spatial concept in the same context (i.e. fuzziness). Existing studies have addressed the fuzziness aspect of the vagueness problem to a great extent, but little work has been done to handle the context-dependency sub-problem. This study focuses on the context-dependency nature of vague spatial concepts. The goal is to enable effective communication of vague spatial concepts in spoken language human-GIS interaction through better managing, sharing, and utilizing contextual knowledge.

Toward this goal, this study has made two major contributions. Firstly, this study provides a Human Communication Framework (HCF) to facilitate our understanding and handling the vagueness problem in human-GIS communication. The HCF explains the vagueness problem in human-human communication, and human communication principles for handling this problem. It helps our understanding about major origins of vagueness, major types of contextual factors, distributed nature of context knowledge and need for building a shared context involved in human-GIS communication. Secondly, this study also provides a collaborative dialogue approach for the GIS to handle the context-dependency problem through collaborative human-GIS dialogues. This approach is driven by the success of collaborative human-human dialogues and distributed nature of context knowledge in human-GIS communication. An agent-based computational model, the *PlanGraph* model, has been developed to support this approach. This model enables the GIS: (1) to build and keep track of the shared context involved in human-GIS communication, (2) to understand the meaning of a vague spatial concept under constraints of the shared context, (3) to repair the shared context and further reduce vagueness through collaborative human-GIS dialogues, and (4) to effectively communicate vague spatial concepts with the user in various contexts.

TABLE OF CONTENTS

Chapter 5 Prototype Software Agent, *GeoDialogue*... 115

Chapter 6 Handling the Context Dependency Problem in Various Situations 137

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

First and most, I would like to thank my advisor, Dr. Guoray Cai, for his immeasurable guidance and support during the course of my graduate study at Penn State. He has taught me how to do research, write papers, give presentations, and everything else about how to become a successful Ph. D. He is an excellent researcher and keeps high standards for himself and his students. I am very grateful to his insights and suggestions that helped me to shape my research skills. Without his guidance and support, I would not be able to accomplish and present the work here.

I am also very grateful to the other Ph. D committee members, Dr. Alan M. MacEachren, Dr. Michael McNeese, and Dr. John Yen. I would like to thank them for serving on my dissertation committees and providing valuable suggestions and other efforts throughout my dissertation study. I feel extremely fortunate to have them on my committee. Their comments and questions on my dissertation study helped me to re-shape presentation of the dissertation work.

I would like to sincerely thank Dr. Alan M. MacEachren. He and my advisor together gave me the opportunity to jump into the amazing research area of natural multimodal GIS. I also learned the Geographical Information Science from him, in particular, about challenges and qualitative research methods in this field, which helped me a lot later to decide the research topic in my study. By working with him together on writing papers/giving talks, I also received lots of help and guidance about how to polish writings and give smooth talks. I would also like to thank him for his generous time and

help on directing me to more literature helpful to my study and polishing my dissertation writing sentence by sentence.

I would also like to give special thanks to Dr. Michael McNeese and Dr. John Yen. Dr. Michael McNeese taught me cognitive science and human-computer interaction, which helped me a lot to understand part of key cognitive concepts and human-GIS interaction involved in my dissertation study. In addition, he also gave me lots of valuable comments on my dissertation research during project group meetings and dissertation related group/individual meetings. His comments always turned out to be very helpful in my study. Dr. John Yen taught me agent-based approaches, including lots of agent-based theories and models that are basis of the agent-based computational model, the *PlanGraph* model, developed to support the collaborative dialogue approach in my dissertation study. I would also like to thank him for allowing me to join his research group discussion about several agent-based theories and models, which helped me to understand these theories better and apply them in the computational work in my dissertation study.

My research benefits a lot from group members in two NSF projects, DAVE_G and GCCM. I learned computational theories and techniques related to GIS from the sub-group led by Dr. Guoray Cai; I learned computational research involved in natural multimodal interfaces from the sub-group led by Dr. Rajeev Sharma; I learned a lot about qualitative research methods involved in the GIScience from the sub-group led by Dr. Alan M. MacEachren; I learned theories and methods used for knowledge elicitation from the sub-group led by Dr. Michael McNeese.

I am deeply indebted to my advisor and Dr. Alan M. MacEachren, who provided me financial support over years. I would also like to express my thanks to the College of Information Sciences and Technology for her support during my graduate study at Penn State. I also thank lots of people, who have helped me in different ways during my time here, e. g. faculties, staff and students in College of Information Sciences and Technology.

As always, the greatest debt owes to my family, in particular, my parents and husband. They always support and encourage me in my endeavors and never doubt my capabilities in my career.

Chapter 1

Introduction

1.1 Motivation

Geographical Information Systems (GIS) are computer based systems for management and analysis of geo-referenced data (Foote & Lynch, 1995; Malczewski, 1999). With over forty years of development and improvement, GIS have been applied to various areas, such as different streams of planning, transportation management, natural resource assessment and management, risk analysis and crisis management, and facilities management.

Most existing commercial GIS adopt the popular direct manipulation interface paradigm, commonly known as windows, icons, menus, and pointing (WIMP) featured interfaces (Egenhofer & Kuhn, 1999). WIMP-based interfaces do not support the interaction between the system and the user efficiently (Frank & Mark, 1991). Such systems are designed for expert users, but not for novice or non-technical users. Users need to be trained how to express their spatial information needs in formal language through WIMP-based interfaces so that the GIS can understand their requests. Users' spatial information needs usually involve spatial concepts. A spatial concept is an abstract mental representation of a category, which refers to a class of geospatial entities, including geospatial objects, relations or processes (Fisher, 2000). A spatial concept in the user's mental representation can be imprecise and is typically qualitative, but the formal language must be precise and usually is quantitative. Therefore, it takes a substantial mental effort for the user to translate his/her imprecise mental representation of the spatial concept into the formal language representation required in GIS commands.

In order to address the problems of WIMP–based GIS and enable wider adoption of the GIS technology by non-technical users (e. g. crisis managers), the GIScience community has paid increasing attention to development of transparent interfaces for GIS (Nyerges et al., 1995), which decrease the difference between the human user's mental models of a problem in an application domain (e. g. crisis management) and those in the GIS tool domain. One stream of such attempts is to introduce natural human communication modalities into human-GIS interaction (Florence et al., 1996; Frank & Mark, 1991; Mark & Frank, 1992a, 1992b; Mark et al., 1987). The assumption in this attempt is that natural modalities, e. g. spoken natural language, gesture/sketch, gaze, facial expression, and body movement, are natural and familiar modalities for human users to express their information needs. Given that human users are more familiar with using natural modalities than using formal language, it will cost users relatively less cognitive effort to convert mental models of spatial concepts into natural modal expressions than formal language-based expressions. As a result, GIS users can focus on doing their domain tasks, instead of spending much time and effort on learning how to use GIS tools.

Natural multimodal interfaces have been developed for geographic information use along with advances in speech processing and capturing human gestures. Early systems with natural modal capabilities can recognize the user's voice command and mouse-simulated pointing gesture (graphics). Examples are "*Put-that-There*" (Bolt, 1980) and CUBISON (Neal et al., 1989). More recent systems adopted pen-based gesture/sketch for interaction of geographical information, such as *QuickSet* (Cohen et al., 1989; Oviatt, 1992, 1996) and *Sketch and Talk* (Egenhofer, 1996). The latest advances in computer vision enable use of natural free-hand gestures in multimodal systems for geographic information access, e. g. Dave_G, which uses both spoken language and free-hand gestures (Cai et al., 2003; MacEachren et al., 2005; Sharma et al., 2003).

1.2 Problem Definition and Scope

Among various natural modalities used in transparent interfaces, spoken natural language was proposed as a major alternative interaction modality for GIS by scholars (Frank & Mark, 1991; Mark & Frank, 1992a; Mark et al., 1987), as it is "the most natural, efficient, flexible and inexpensive" modality for everyday human communication (Zue & Glass, 2000) and it naturally fits to human mental models of spatial concepts (Frank & Mark, 1991). To develop a usable spoken language enabled GIS, one of the challenges that must be addressed is how to semantically interpret a spoken language request involving spatial concepts. Natural language messages do not commonly have one-to-one mapping into formal semantics. In particular, the user's natural language requests can be incomplete, ambiguous, vague, and inconsistent (Bosc & Prade, 1993; Owei, 2002; Robinson et al., 1985; Wang, 2003). Existing studies have made much progress on handling the problems of incompleteness, ambiguity, and inconsistence (Bosc & Prade, 1993; Owei, 2002; Robinson et al., 1985; Wang, 2003), but it is still an open problem how to handle the vagueness problem in human-GIS communication, although some initial progress (Duckham et al., 2001; Kollias & Voliotis, 1991; Wang, 1994, 2000, 2003; Yao & Thill, 2005, 2006) has been achieved.

The meaning of a spatial concept involved in human-GIS communication through natural modalities, particularly, through natural language, can be fuzzy and context-dependent. The naturally formed categories that spatial concepts refer to usually do not have clear boundaries (Barsalou, 1987, 2000; MacEachren, 1995). Such unclear boundaries are called "fuzzy boundaries" in the geography field (MacEachren, 1995), or "graded structures" in the cognitive science field (Barsalou, 1987). The unclear boundary of a naturally formed category leads to the unclear meaning of the concept referring to that category, that is, the inherent fuzziness property of the concept. According to Barsalou's study results (Barsalou, 1987), given different contexts, humans may use different criteria to determine the grade of membership of each member in a category; given the same context, different individuals have different mental constructions of the same concept. As a result, the context-dependency property of a naturally formed category leads to the context-dependent meaning of the concept referring to that category, that is, the context-dependency property of the concept. Therefore, the meaning of a naturally formed spatial concept usually holds these two properties, inherent fuzziness and context-dependency. In this study, such a spatial concept is defined as a "vague spatial concept." These two properties of vague spatial concepts lead to the vagueness problem in human-GIS communication.

The goal of this study is to facilitate spoken natural language enabled GIS to handle the vagueness problem. Existing GIS can understand precise and quantitative requests only, while the user's spoken natural language requests involving vague spatial concepts are vague and qualitative. To enable successful human-GIS communication involving vague spatial concepts, we must meet two fundamental challenges here due to the two unique characteristics of vague spatial concepts. The first challenge is to enable the GIS to handle the inherent fuzziness problem, that is, to enable the GIS to convert the fuzzy meaning of a vague spatial concept into a deterministic one or represent it fuzzily (Yao & Thill, 2006). Numerous studies (Cross & Firat, 2000; Guesgen & Albrecht, 2000; Kollias & Voliotis, 1991; Wang, 1994, 2000, 2003) have been performed to facilitate handling the fuzziness problem. Approaches developed in these studies provide only a best guess of fuzzy meanings of vague spatial concepts communicated by the user, which may not be what the user intended. The second challenge is to enable the GIS to handle the context-dependency problem, that is, to enable the GIS to determine a meaning of the vague spatial concept which matches the current context involved in communication. In the GIScience community, only a few studies, e. g. Wang's studies (Wang, 1994, 2000, 2003) and Yao's studies (Yao & Thill, 2005, 2006), have started to address the context-dependency problem in human-GIS communication. However, the approaches of these studies do not support the GIS to handle the context-dependency problem effectively.

1.3 Research Objectives

This study focuses on handling the vagueness problem, particularly, the context-dependency problem, involved in spoken natural language human-GIS communication of spatial concepts. The research objective of this study is to enable effective communication of vague spatial concepts in spoken language human-GIS interaction through better managing, sharing, and utilizing contextual knowledge.

This study has made two major contributions toward the research goal. Firstly, a Human Communication Framework (HCF) is established to facilitate our understanding and handling the vagueness problem in human-GIS communication. This framework explains the vagueness problem in human-human communication and human communication principles involved in handling the vagueness problem, in particular the context-dependency problem. This framework facilitates our understanding about major origins of vagueness, major types of contextual factors, distributed nature of context knowledge and the need for building a shared context involved in human-GIS communication.

Secondly, a collaborative dialogue approach is proposed and developed in this study. This approach is driven by the success of collaborative dialogues in human-human communication and the distributed nature of context knowledge in human-GIS communication. An agent-based computational model, the *PlanGraph* model, is built to support this approach. This model includes two components, the *PlanGraph* and associated reasoning algorithms. The *PlanGraph* represents the dynamic knowledge kept by the GIS on the shared context between the user and the system. Associated reasoning algorithms support the GIS to build and keep track of the shared context with use of the *PlanGraph* as the dialogue proceeds. They also support the system to repair the shared context and further reduce vagueness through collaborative human-GIS dialogues. Thus, the GIS can understand the meaning of a vague spatial concept under constraints of the shared context. Finally, this model also enables the GIS to reach a shared understanding about the vague spatial concept with the user in various situations where the system and the user have different understandings about the current context involved in human-GIS communication.

1.4 Organization of the Thesis

The remainder of this thesis is organized as follows. Chapter 2 reviews existing studies related to handling the two sub problems, the fuzziness problem and the context-dependency problem, of the vagueness problem in human-GIS communication. Details of the HCF are provided in Chapter 3. This framework includes three components, the human conceptual system, the human geospatial knowledge acquisition process, and the human communication process. It facilitates our understanding of the vagueness problem in human-human communication, in particular the context-dependency problem. It also shows effective human solutions for handling the vagueness problem, that is, collaborative dialogues. Chapter 4 describes details of the collaborative dialogue approach developed in this study in detail. This approach is designed under the human emulation principle. It helps the GIS to handle the context-dependency problem through collaborative human-GIS dialogues by emulating a human agent's reasoning and behavior. It is supported by the agent-based computational model, the *PlanGraph* model. Chapter 5 describes a prototype software agent, *GeoDialogue*, which incorporates the collaborative dialogue approach and the *PlanGraph* model. Implementation details of the *PlanGraph* model in this agent are also explained in this chapter. In Chapter 6, various situations involving the context-dependency problem in human-GIS communication are categorized and discussed. Several sample human-*GeoDialogue* dialogues are explained to illustrate how the collaborative dialogue approach and the *PlanGraph* model can help the GIS to handle the context-dependency problem in various situations. Finally, Chapter 7 concludes this thesis with a discussion of the major contributions, comparisons with previous studies, and future work.

Chapter 2

Existing Studies

This study is concerned with supporting the GIS to handle the vagueness problem involved in human-GIS communication of spatial concepts. As introduced in Chapter 1, two challenges are met for development of spoken natural language enabled GIS, including handling the inherent fuzziness problem and handling the context-dependency problem. This chapter reviews existing studies related to handling these two challenges, respectively.

2.1 Overview

The GIScience community has recognized the importance of handling the vagueness problem for many years (Duckham et al., 2001; Erwig & Schneider, 1997; Frank & Mark, 1991; Mark et al., 1987; Montello et al., 2003; Robinson, 1990, 2000). However, there is no common understanding about vagueness of spatial concepts in the GIScience field. One view (Couclelis, 1996; Duckham et al., 2001; Erwig & Schneider, 1997; Montello et al., 2003; Worboys, 1998) states that vagueness of spatial concepts is subject to the sorites paradox (Hyde, 2005) as well as dependent on the context involving these concepts. For example, Worboys (2001a) defined vagueness as "imprecision in concepts used to describe the information (e.g. *near* and *far* are vague metric properties of a spatial object describing distance from the observer)", and pointed out that vagueness is "often have associations with context-dependency and subjectivity”. The other view focuses on only the inherent fuzziness aspect of spatial concepts. For example, Fisher (2000) conceives of vagueness from the philosophical view by applying the sorties paradox rule to test whether a spatial concept is vague or not. His work does not mention the context-dependency property of vague spatial concepts.

In this study, we take the view on vagueness of spatial concepts, which addresses both the fuzziness aspect and the context-dependency aspect of spatial concepts. Thus, two challenges are involved in handling the vagueness problem. The first challenge is how to handle the inherent fuzziness problem, and the second challenge is how to handle the context-dependency problem.

The Inherent Fuzziness Problem

The GIS needs to convert the meaning of a vague spatial concept into a metric measure needed in certain GIS commands or represent it fuzzily in order to satisfy the user’s request involving that concept. In the conversion process, the system needs to determine a crisp boundary for that concept, which has a fuzzy boundary. However, the process to determine the boundary of a vague spatial concept subjects to the sorties paradox (Erwig & Schneider, 1997; Fisher, 2000; Montello et al., 2003; Varzi, 2001). For example, in a given context where the user is looking for grocery stores *near* his/her residence location for the purpose of shopping for food by driving a car in the small town, State College, the system can determine that groceries stores within two or three miles are near to this user, and that those beyond seven or eight miles are far to this user.

However, it is hard for the system to determine whether the grocery stores between three miles and seven miles are near or not.

The Context-Dependency Problem

The meaning of a vague spatial concept is context-dependent, and varies in different contexts, e. g. to different users (Worboys, 2001a), for different tasks (Erwig & Schneider, 1997), and in different spatial contexts (Robinson, 2000; Worboys, 2001b).

The task context defined in this study includes the user's goals, associated activities towards these goals, and other properties associated with these activities involved in or to be involved in the event involving the vague spatial concept. The task contextual factor can be the final goal of the event, actions performed or to be performed in the activities in the event, or other properties associated with these actions (Erwig & Schneider, 1997; Yao & Thill, 2005). When a user requests a map to be displayed by a GIS, it is often because the user is participating in or is going to participate in an event. The user's final/sub goals in this event and facilities to be involved in actions in this event influence the user's understanding about the vague spatial concept (Egenhofer & Shariff, 1998; Erwig & Schneider, 1997). For example, a person *A* has just moved to a new city and needs help to find restaurants for his daily lunch as well as grocery stores for his weekly food shopping purpose *near* his current apartment. The same concept *near* used by the same person *A* most likely means different distance relations for these two different purposes. A may consider five miles of driving distances as *near* for the food shopping purpose, but too far for the lunch purpose. For a same purpose, different transportation modes (Yao & Thill, 2005) will also lead to different distance relations for the same person in the same spatial context. For instance, A considers grocery stores about five miles away from his residence location as *near* when planning to buy food by driving, while takes them as far when planning to walk.

The spatial context is defined in this study as spatial locations where the event involving the vague spatial concept takes place or will take place. There is evidence that the meaning of a vague spatial concept, such as "*near"*, is dependent on the spatial context involving the spatial concepts (Fisher & Orf, 1991; Lloyd & Heivly, 1987; Robinson, 2000; Worboys, 2001a). For example, concerning the vague spatial concept *near*, the spatial context involves the spatial scale/size of area covering all spatial distances considered, topological closeness or reachability of routes, intervening opportunities of routes (Yao & Thill, 2005), relative spatial distances, and absolution spatial distances (Gahegan, 1995), etc. For example, the meaning of *near* is different in different spatial contexts (Figure **2-1**) for the same user, Bob, and the same task, the food shopping task. By driving, Bob understands all grocery stores within the red rectangle area as *near* when considering the spatial area of the Palm Beach county as the spatial scale of the food shopping event, while he regards part of these stores as far when considering the red rectangle area as the spatial scale of the shopping event. The grocery store A and B in Figure **2-1** are both nine miles away from Apartment A. However, there is no bridge cross the river between Apartment A and the grocery store A. Thus, the road distance between Apartment A and the grocery store A is much farther than that between Apartment A and the grocery store B. The road distance between Apartment A and the grocery store C is same as that between Apartment A and the grocery store D. However, the intervening opportunities between Apartment A and the grocery store D are twice of that between Apartment A and the grocery store C. Thus, Bob considers the grocery store C as *near*, but regards the grocery store D as far. When considering all grocery stores in Palm Beach, Bob takes all grocery stores in the red rectangle area as *near* because the spatial distances between these stores and Apartment A are relatively shorter than the others. When Bob is going to buy food by walking, he needs to consider the absolute spatial distance between each grocery store and his apartment. In this situation, he considers the stores within one mile of walking distances of his apartment as *near*, and these in the red rectangle area beyond two miles as far.

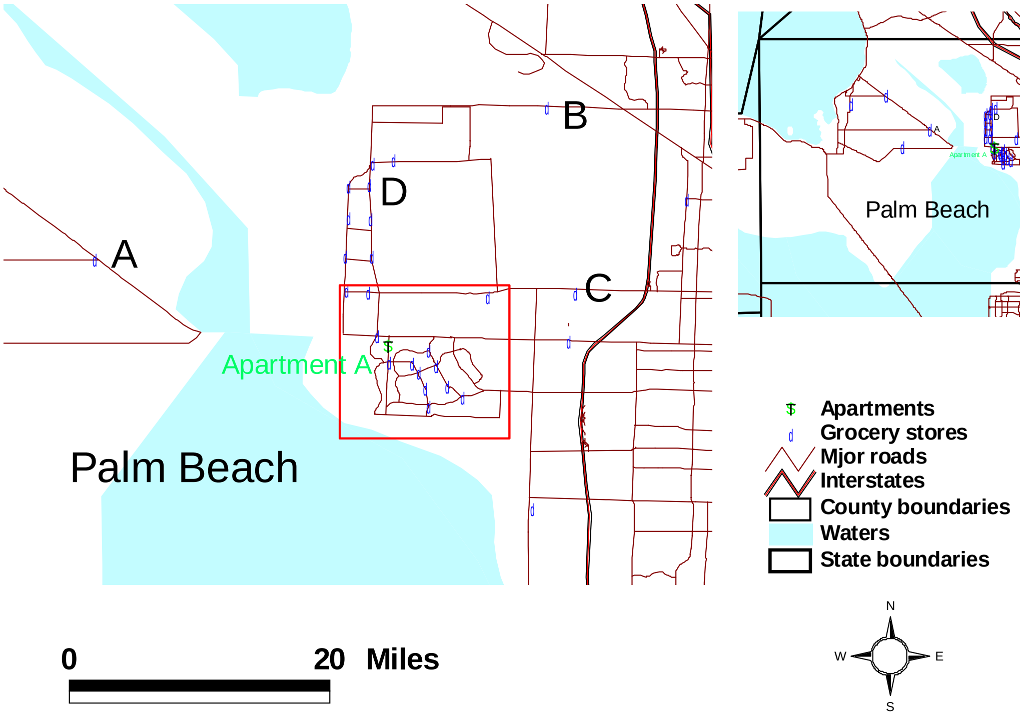


Figure **2-1**: Example Spatial Contexts

Different individual users can understand the same vague spatial concept differently and the same user may understand the same concept differently at different time (Barsalou, 1987). For example, Worboys' work (2001a, 2001b) and Robinson's experiments (2000) on the *near* spatial relation indicate that different people have different understandings of the same spatial distance relation. This is because individual users may differ in cultural and education background, personal experiences, or other aspects (Yao & Thill, 2005). In this study, this type of contextual factors is simply referred to as the human context, similar to the term "subjectivity" in Worboys' work (2001a).

The remainder of this chapter reviews existing studies related to handling these two challenges, respectively.

2.2 Handling the Inherent Fuzziness Problem

In the GIScience field, existing studies related to handling the inherent fuzziness problem can be divided into three categories (Table **2-1**). The first category of studies concentrates on development of formal models to model the inherent fuzziness of spatial objects/relations in spatial databases. The second category focuses on the development of experimental studies which model the fuzziness of vague spatial concepts. The third category works on development of formal models to support human-GIS communication involving vague spatial concepts. This section reviews these three categories of studies respectively.

Table **2-1**: Studies for handling the inherent fuzziness in human-GIS interaction

Study Category	Studies	Focus
Development of formal models for fuzziness in spatial database	(Burrough & Frank, 1996; Erwig & Schneider, 1997; Leung, 1982, 1987)	Development of formal models modeling geospatial objects/regions, whose purpose is for geospatial data processing
	(Cohn & Gotts, 1996; Krishnapuram et al., 1993; Zhan, 1997, 1998)8)	Development of formal models modeling geospatial relations, whose purpose is for geospatial data processing
	(Burrough et al., 2000) (Galton, 2003; MacMillan et al., 2000; Stell, 2003)	Development and application of formal models for geospatial data analysis
Development of experimental studies which model the fuzziness of vague spatial concepts	(Fisher & Orf, 1991; Robinson, 1990, 2000; Worboys, 2001b; Zhan, 2002) (Montello et al., 2003)	Development of experimental studies to acquire fuzzy membership functions for vague spatial concepts
Development of formal models for fuzziness in human-GIS communication through natural language enabled interfaces	(Cross & Firat, 2000; Guesgen & Albrecht, 2000; Kollias & Voliotis, 1991; Wang, 1994)	Development of formal models which support queries containing vague terms (conversion of vague-terms to non-vague GIS conditions)
	(Wang, 2000, 2003)	Development of natural language interfaces which support communication of vague spatial concepts through natural language

2.2.1 Modeling Fuzziness of Spatial Objects/Relations in Spatial Database

Numerous studies in the GISience field have been performed to handle the inherent fuzziness problem in geospatial data processing (Table **2-1**), such as studies about modeling fuzzy spatial object/region boundaries (Burrough & Frank, 1996; Erwig & Schneider, 1997; Leung, 1982, 1987), studies modeling the fuzziness in spatial relations (Cohn & Gotts, 1996; Krishnapuram et al., 1993; Zhan, 1997, 1998), and studies about geospatial data analysis applied to fuzzy data (Burrough et al., 2000; Galton, 2003; MacMillan et al., 2000; Stell, 2003).

The studies modeling fuzzy regions (Burrough & Frank, 1996; Cohn & Gotts, 1996; Erwig & Schneider, 1997; Leung, 1982, 1987; Wang & Hall, 1996) developed mathematical models to represent geospatial objects with fuzzy boundaries. For example, Cohn and Gotts (1996) presented an "Egg-Yolk" model to represent fuzzy regions. In this model, the whole region with a fuzzy boundary is represented as *egg*, the exact and determined portion of the region as *yolk* and indeterminate parts as *white*. Such mathematical models can be used to process spatial data representing spatial objects/regions with fuzzy boundaries, and further to serve for the purpose of human-GIS communication.

Similarly, the studies modeling fuzzy spatial relations also developed mathematical models to represent spatial relations with fuzzy boundaries based on the fuzzy set theory. For example, Zhan (1997,1998) developed a fuzzy representation of spatial regions with fuzzy boundaries and methods for computing spatial relations between spatial objects with fuzzy boundaries based on the fuzzy set theory. Krishnapuram and his colleagues (1993) developed methods based on the fuzzy logic to compute the directional relations between spatial regions with fuzzy boundaries. These models can be used to process spatial data representing spatial relations with fuzzy boundaries, and further show the fuzzy boundaries of spatial relations to the user.

Classification studies are typical examples in the studies about development and application of formal models for geospatial data analysis. For example, Burrough, and his colleagues (2000) developed a fuzzy k-means approach to classify landforms based on statistical sampling results of DEM data. MacMillan and his colleagues (2000) developed a model to segment landforms into landform facets. This model used derivatives computed from DEM data and a fuzzy rule to identify up to 15 morphologically defined landform facets. Galton (2003) investigated granularity-sensitivity, and proposed a framework, which guides investigation of granularity-sensitivity in a “more rigorous, systematic, and general way”.

2.2.2 Experiments on Modeling Fuzziness of Vague Spatial Concepts

To understand what a vague spatial concept is, GIS must have specific knowledge about the degree to which the user believes each member in that category belongs to that category. The fuzzy set theory (Zadeh, 1965, 1978, 1983a, 1983b) has been applied to model such kind of knowledge about the inherent fuzziness property of the vague spatial concept (Fisher & Orf, 1991; Montello et al., 2003; Robinson, 1990, 2000; Worboys, 2001b; Zhan, 2002).

Fisher and Ort (1991), Worboys (2001b), and Montello et al. (2003) all work on modeling the inherent fuzziness aspect of vague spatial concepts in real environments. In Fisher and Ort’s study (1991), the human subjects were asked if they agreed with a computer program’s initial guess about whether particular buildings on the campus were *near/close* to a central building on the campus, and then they were asked to modify the initial guess if they disagreed. A fuzzy membership function for each subject’s understanding of *near/close* relations was established based on their answers. In Worboy’s study (2001b), two groups of subjects on the campus were asked to pick up places which are *near* or not *near* from a list of places on the campus. The results from the experiment were analyzed by four different approaches, including the three-valued logic, fuzzy membership functions and higher valued logic. Montello and his colleagues (2003) investigated the fuzziness property exhibited on the boundary of a downtown area. Pedestrians around *Downtown Santa Barbara* were asked to delineate the area on a map which they think is the downtown area, the area which they are 100% and 50% confident is downtown, and the representative location in downtown. The participants' answers were used to analyze the inherent fuzziness nature of the downtown area.

Robinson (1990, 2000) and Zhan (2002) both investigated the fuzziness aspect of spatial concepts in a geographical context/on a map. Robinson (1990) describes how a computer program could be used to acquire the fuzzy membership function of the vague spatial concept *near*. The human subject was first asked by the computer program if he/she agrees with the system's initial guess on the fuzzy membership value of a city, and if not, he/she needs to modify the initial guess. Each individual's answers were used to construct the individual's conception about *near*. In a subsequent study, Robinson (2000) uses this program to acquire the user's fuzzy membership functions on *nearness* and *farness* in two different geographical contexts, respectively. Multi-person's fuzzy membership functions were explored in the study. Zhan (2002) investigated fuzziness in several vague spatial concepts related to the topological relation, *overlay*. In his study, human subjects were interviewed about their judgments on the *overlay* relation between two overlaid diagrams. Their answers were used to establish fuzzy membership functions for the topological relation concept *overlay* described by "a little bit, somewhat, and nearly completely". The spatial concept *overlay* described by each of these terms is vague. The fuzzy membership functions (Figure **2-2**) for fuzzy sets representing different degrees of "cover a little", "cover somewhat" and "nearly completely cover" were obtained from 32 subjects.

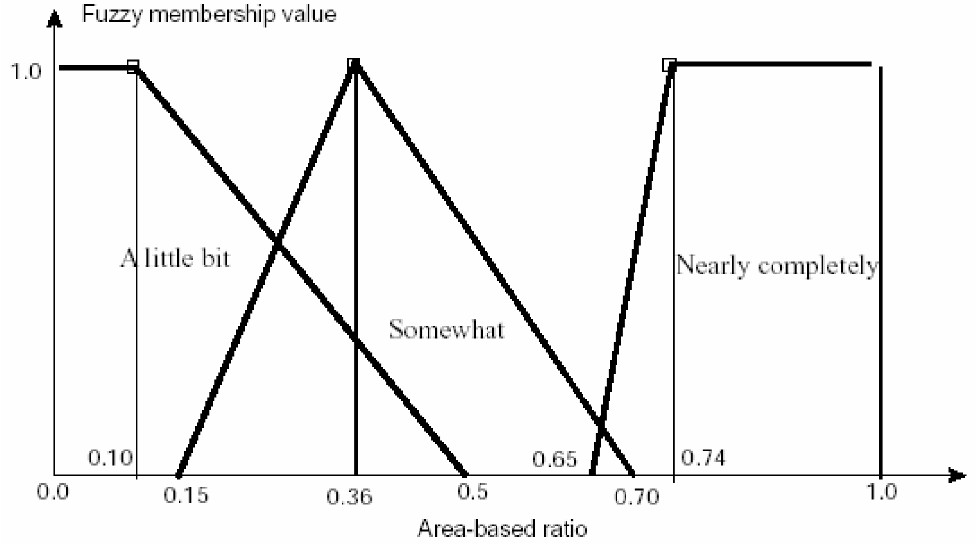


Figure **2-2**: Fuzzy Membership Functions for *Overlay* (Zhan, 2002)

2.2.3 Fuzziness of Vague Spatial Concepts in Human-GIS Communication

Most existing studies modeling fuzziness of spatial concepts involved in human-GIS communication (Cross & Firat, 2000; Guesgen et al., 2003; Kollias & Voliotis, 1991; Wang, 1994, 2000, 2003) have developed formal models based on the fuzzy set theory and fuzzy logic. The purpose of these studies is to support the GIS to convert fuzzy meanings of vague spatial concepts in user queries into non-fuzzy measures.

Guesgen & Albrecht (2000), Cross and Firat (2000), Kollias and Voliotis (1991), and Wang (1994) have investigated formal models for fuzziness of spatial concepts in human-GIS communication through Structured Query Language (SQL) based queries.

Guesgen & Albrecht (2000) proposed a conceptual scheme to integrate the fuzzy set with the GIS. This scheme was used to model qualitative spatial relations among spatial objects in the GIS. Their study did not show how such a scheme could be implemented and applied for human-GIS communication of vague spatial concepts. Cross and Firat (2000) and Kollias and Voliotis (Kollias & Voliotis, 1991) focused on development of a fuzzy object data model or fuzzy relational database model that would enable the GIS to execute SQL queries involving vague spatial concepts. In their studies, the GIS database structure needs to be modified. To establish such a model for a vague spatial concept, the user needs to define the fuzzy membership for that concept first either through the computer screen or directly in the database. However, the user may not be able to clearly define such a function. In addition, the user still needs to learn how to write the SQL query before using the system. Wang (1994) studied an approach which enables the GIS to convert vague spatial concepts in the user's query into non-fuzzy conditions in GIS commands. The GIS database does not need to be modified with his approach as it does in the two studies by Cross and Firat (2000) and Kollias and Voliotis (Kollias & Voliotis, 1991). However, the fuzzy membership function in his study is pre-built before a user's interaction with the system by his approach. Consequently, the estimated meaning of a vague spatial concept based on the pre-built fuzzy membership function may not be same as the meaning of that concept that a specific user intends to communicate, and the user may not agree with the system's estimation of the boundary limit of the vague spatial concept.

Wang (2000, 2003) further extended his 1994 study to support human-GIS communication of vague spatial concepts through natural language. Wang (2000) explored his approach in 1994 for the GIS to handle the ambiguity problem and the inherent fuzziness problem in natural language queries based on the possibility theory and the fuzzy set theory. Systems with this approach handle the ambiguity problem by estimating the most possible parsing tree of the user's natural language query and defuzzy a vague concept by estimating the most possible meaning of the vague spatial concept involved. Wang (2003) further extended his 2000 study. The three-step technique developed in this study enables the GIS to handle three types of uncertainties involved in human-GIS communication, incompleteness, ambiguity, and fuzziness. The first step is to estimate the most possible missing component in the user's natural language query. The second step is to estimate the most possible parsing tree in the case that there are multiple parsing results corresponding to the user's natural language query. In the third step, during the translation process, the system defuzzies the fuzzy expression (if possible) into the most possible deterministic condition.

2.3 Handling the Context-Dependency Problem

The GIScience community has recognized the importance of the context to the meaning of a vague spatial concept for many years (Ahlqvist, 2000; Briggs, 1973; Day, 1976; Erwig & Schneider, 1997; Fischer et al., 1991; Kaplan, 1973; Kunnapas, 1960; Lloyd & Heivly, 1987; Worboys, 2001a). Sharma et al. (1994) pointed out that human processing and reasoning of qualitative properties of the spatial data is context-dependent, and so, we need a model to explicitly map the relationship between the context and the metric properties of data. Gahegan (1995) proposed to contextualize the proximity relation with various contextual factors, including the scale and size of the area being considered, the attractiveness of objects, spatial distribution of objects, reachability (topological closeness or network distance). Hernandez et al. (1995) also pointed out that the proximity concept is context-dependent. They proposed a framework built on three elements to establish a proximity spatial relation. These elements include two spatial objects between which the proximity spatial relation is studied – named the primary object (PO) and the reference object (RO), and the frame of reference (FofR). The FofR is instrumental in setting an upper bound on all distances within the area under consideration. It is also used to formalize different types of contextual factors as to the proximity relation.

Although numerous studies have recognized the role of the context on affecting the meaning of a vague spatial concept, only a few studies (Wang, 1994, 2000, 2003; Yao & Thill, 2005, 2006) have started to address the context-dependency problem involved in human-GIS communication of spatial concepts.

Wang (1994, 2000, 2003) studied the role of the context on the meaning of a vague concept involved in the user's SQL query or natural language-based request. He (1994) gave several examples about how the GIS using his approach can adjust its inference on the meaning of a vague spatial concept depending on contextual information provided by the user. In these examples, the GIS with his approach can consider only one major contextual factor's influence on each of the vague concepts. For example, the GIS considers only the user's transportation mode's influence on the meaning of the vague spatial concept, *near*, and only the influence from the spatial area on the meaning of *high*/*low* describing the temperature. In case the major contextual information considered by the GIS is not available in the user's SQL query or natural language request, the GIS will use the default value (a guessed value) of that contextual factor to estimate the meaning of the vague concept. However, the guessed contextual information may not be true. As a result, the estimated meaning of the vague spatial concept may not be agreed upon by the user. In addition, it is not clear how the major contextual factor is modeled in the GIS and kept track of by the GIS.

Yao and Thill (2005) explored a statistical approach, Ordered Logit Regression, to computationally model the context-dependency property of proximity relations and reported an empirical study for constructing a context-contingent model for proximity spatial relations, e. g. *near*, and *far*. A computational context-dependent model of meanings of proximity relations was built by implementing this approach so that the GIS could translate metric distance measures (e. g. 5 miles of road distance) into linguistic distance measures (e. g. *near* or *far*). The contextual factors included in the computational model of proximity relations include the scale effect, type of activities planned at the destination, topological closeness or reachability, intervening opportunities, transportation mode, the familiarity level with areas and routes, financial and time budget, and personal/demographical characteristics. This model was also used to determine whether and how each contextual factor influenced the human subject's perception of proximity relations. Their empirical study was conducted on the campus of the University of Buffalo. Undergraduate students were asked for their judgments on several proximity relations, including *very near*, *near*, *normal*, *far*, and *very far*, in 19 hypothetical scenarios in Buffalo. The computational model built based on the empirical study revealed that several contextual factors, including the metric network distance, the level of familiarity with the area in the vicinity of the destinations, the transportation mode, and the type of activities, significantly influenced the students' perception of the proximity relations. This study also indicated that the result of the computational model predicts the linguistic distance well given the metric distance, but the reverse prediction does not work well. This means that it is not suitable to use the result of this model to translate linguistic distance (near or far) into metric distance given contextual information.

Recently, Yao and Thill (2006) proposed a conceptual framework (Figure **2-3**) for geospatial information systems to handle qualitative terms or queries based on their earlier work. They considered the linguistic terms representing vague spatial concepts as typical examples of the qualitative terms, whose meanings are context-dependent. Qualitative spatial reasoning about such terms usually involves handling the context-dependency problem. Their conceptual framework was described by three modules, the *Spatial Database* module, *Language Interpreter* and *Qualitative Spatial Reasoner.* The *Language Interpreter* decomposes a user query into several types of elements needed in formal queries to GIS and translates fuzzy linguistic terms. If qualitative terms are found, then this module sends the interpreted data to the *Qualitative Spatial Reasoner* module; otherwise, this module directly passes the data to the *Spatial Database* module. The *Qualitative Spatial Reasoner* consists of the module of Central Control Unit (CCU) and multiple qualitative reasoning models for each particular type of qualitative spatial relations. CCU controls the information flow in the reasoner module, communicates with the spatial database, and synthesizes results from the reasoning models and/or from the spatial database. In this framework, the system can communicate with the user to clarify uncertainties involved in the user query or the information (e. g. context information) needed in various qualitative reasoning models in the reasoner module. Their study also provides a prototype of the *Qualitative Spatial Reasoner* to illustrate this conceptual framework. This study is valuable for proposing a framework for handling the context dependency problem involved in qualitative reasoning about spatial relations. However, it has some limitations. In this framework, the context information is directly stored in context-contingent models of qualitative spatial relations. The framework does not include a model or module to enable the system to keep track of the dynamic context involved in human-computer interaction.

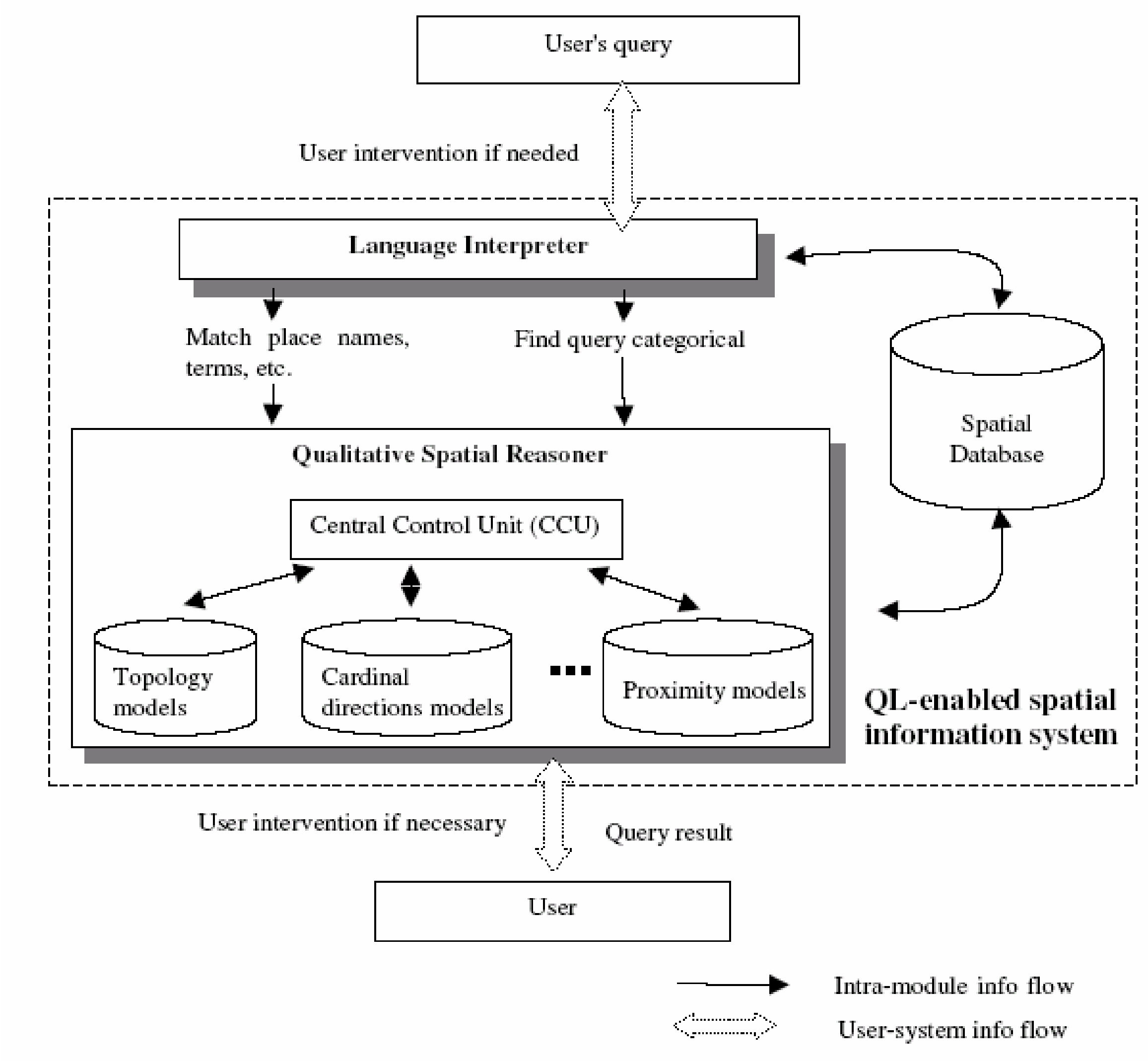


Figure **2-3**: Conceptual Framework for Qualitative Reasoning (Yao & Thill, 2006)

2.4 Summary and Discussion

This chapter reviews existing studies in the GIScience field related to handling the two challenges involved in human-GIS communication of vague spatial concepts. Most existing studies focus on handling the inherent fuzziness problem. Many studies have discussed the role of the context in human-GIS communication of vague spatial concepts or human's qualitative reasoning. However, only a few studies have started to consider how to handle the context-dependency problem in human-GIS communication.

Formal models based on the fuzzy set theory and the fuzzy logic have been developed in existing studies to model fuzziness of vague spatial concepts. The results of these models can provide only a best guess to the meaning of a vague spatial concept involved in human-GIS communication, and leave no room for the user to clarify and confirm the estimated result generated by the system. Therefore, they do not ensure that the GIS and the user reach a shared understanding about the fuzzy boundary of the vague spatial concept communicated. In addition, the contexts where these models were constructed are fixed or it is assumed that "geographic reality is certain, crisp, unambiguous, independent of context and capable of quantitative representation" (Duckham et al., 2001). This assumption may not be true. In particular, during human-GIS communication of vague spatial concepts, these models can help to develop strategies that allow the system to have a guess about the meaning of a vague spatial concept communicated with the user, but they do not ensure that the communication is successful.

As reviewed above, a significant body of literature has recognized contextual influences on the meaning of the vague spatial concept. However, only a few studies have started to address how to handle the context-dependency problem. These studies do not have comprehensive understanding about how the context plays a role in human-GIS communication of vague spatial concepts, neither offer a way to computationally model the context involved in human-GIS communication. There are no workable proposals on how to keep track of the context. Therefore, they do not effectively support the GIS to infer the meaning of that concept with matching the current context involved in communication. In addition, these studies also just provide a best guess to the meaning of the vague spatial concept involved based on contextual information available or inferred, and their results do not ensure that the GIS can reach a shared understanding with the user about the meaning of that concept in various contexts.

In summary, there has been lack of effective handling of the vagueness problem, in particular, the context-dependency problem. Thus, this presents an obstacle for developing spoken natural language enabled GIS.

Chapter 3

Human Communication Framework

By following Frank (1993) and Mark (1993), we believe that a better matching with human cognition would lead to a better interface, and better matching with human cognition is always one of the goals toward development of transparent interfaces. Under this assumption, we believe that a deeper understanding about how the vagueness problem is generated and handled in human communication may motivate solutions for spoken natural language human-GIS communication of vague spatial concepts. Toward this goal, we have developed a Human Communication Framework (HCF) in this study.

Human communication is a process of making sense of the world and sharing that sense with others (Leeds-Hurwitz, 1989; Tubbs & Moss, 2000). In interpersonal communication, that is, human-human communication, most communication messages are transmitted by spoken language, along with other nonverbal messages, such as visual contact, gestures, facial expression, and posture (Heath & Bryant, 2000; Tubbs & Moss, 2000). What is communicated in human-human communication are messages, not meanings of the messages (Cherry, 1978; Heath & Bryant, 2000; Tubbs & Moss, 2000; Verderber, 1987). The meaning of a message is generated by the human conceptual system (Barsalou, 1982, 1987, 1999, 2000, 2003; Jackendoff, 1991; Landau & Jackendoff, 1993). The geospatial knowledge used in the human conceptual system to generate the meaning of a spatial concept is formed through human's knowledge acquisition process (Egenhofer & Mark, 1995; Golledge, 1991, 2002; Kuipers, 1978;

Tversky, 1993). Based on these previous studies, we describe the cognitive process in the HCF, which involves three components: the geospatial knowledge acquisition process, the human conceptual system, and the human-human communication process (Figure **3-1**).

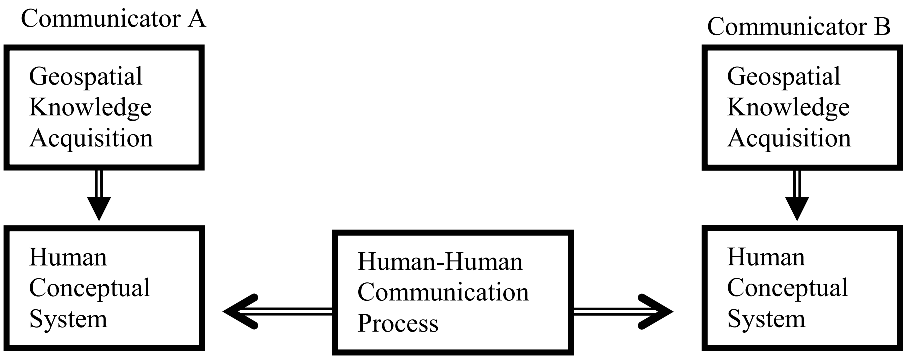


Figure **3-1**: Human Communication Framework

Details of the three components of the HCF are provided in Sections 3.1, 3.2, and 3.3, respectively. Then, origins of vagueness and human solutions in human-human communication are summarized in Section 3.4 and Section 3.5, respectively.

3.1 Human Conceptual System

A human conceptual system provides knowledge about the world for the human. It plays central roles in the reasoning involved in human communication, in particular, about meanings of concepts. We need to understand how the context-dependent meaning of a vague spatial concept is generated by the human conceptual system. Four major types of theories (Barsalou, 2003) have been proposed for the human's conceptual system, including semantic memory, exemplar models, feed-forward connectionist nets, and the situated simulation theory. The situated simulation theory (Barsalou, 2003) emphasizes the contextualized characteristic of the conceptual system; that is, the meaning of the concept is not static, but constructed in terms of the current need/context/situation. Therefore, the human conceptual system described in the HFC mainly follows the situated simulation theory developed by Barsarlou (Barsalou, 1982, 1987, 1999, 2000, 2003). According to the perceptual symbol theory (Barsalou, 1999), which is part of the situated simulation theory, human knowledge on a concept is stored as perceptual symbols in Long Term Memory (LTM). The perceptual symbol corresponding to a concept is used to generate conceptual simulations in Short Term Memory (STM) on the fly constrained by simulation of the current context (Figure **3-2**). A concept is a simulator which can generate infinite specific simulations of the concept to fit to different contexts.

A perceptual symbol is generalized from the extraction of a subset of the perceptual state that a person perceives from an information source (Figure **3-2**). The "subset" of the perceptual state is what the person has paid attention to. Perceptual symbols referring to the same concept are all associated with that concept. However, due to the graded structure of a category (Barsalou, 1987), the boundary between perceptual symbols belonging to a concept and those not belonging to that concept also exhibits the graded structure, that is, the inherent fuzziness property. The perceptual symbols are multimodal from the five human senses, including vision, audition, haptic, olfaction and gestation. Therefore, there may be multiple perceptual symbols from multimodalities corresponding to the same concept, e. g. words and images (Paivio, 1971; Potter & Faulconer, 1975).

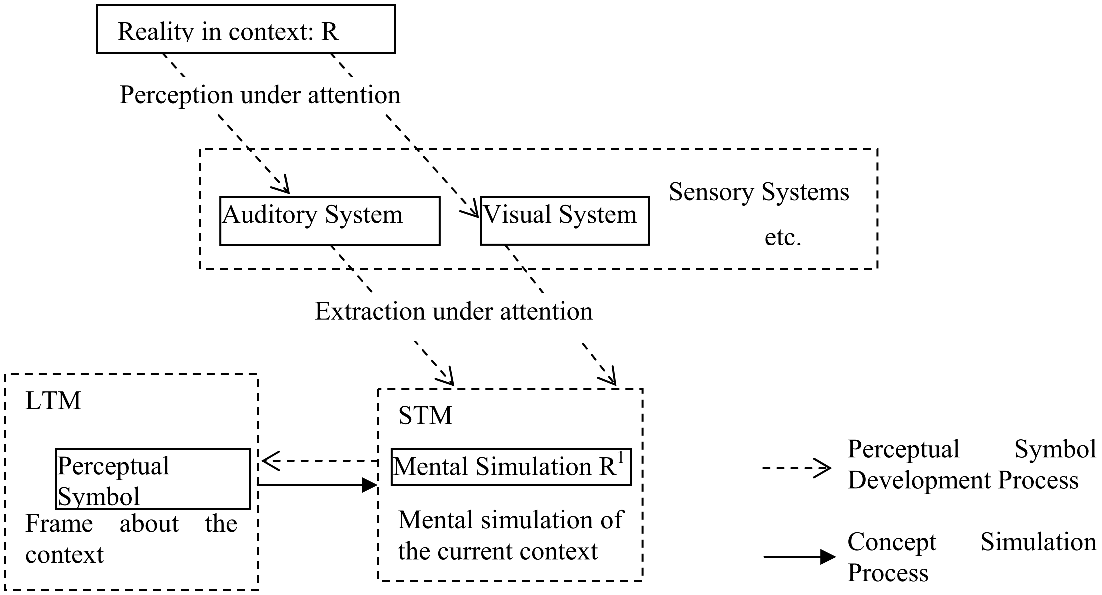


Figure **3-2**: Structure of the Human Conceptual System

Perceptual symbols referring to different components of the same concept are integrated into a *frame* in terms of relationships among these related perceptual symbols. The *frame* has four basic properties (Barsalou, 1999; Barsalou & Hale, 1993; Barsalou & Hutchinson, 1987): 1) predicates – corresponding to unspecified sub-regions in the frame; 2) attribute-value bindings – corresponding to specific instantiations of perceptual symbols under sub-regions in the frame; 3) constraints – corresponding to the inherent relationships among different perceptual symbols; and 4) recursion structures.

The simulation of a concept, that is, a specific meaning of that concept, is generated through instantiation of its corresponding perceptual symbol/frame. The simulation of the concept is highly context-dependent. A perceptual symbol/frame corresponding to a concept is always instantiated under partial instantiation of another frame representing the context which involves that concept (see Figure **3-2**) (Barsalou, 1999; Carey, 1985; Fillmore, 1985; Keil, 1989; Lehrer, 1974). The result of the instantiation of the perceptual symbol/frame is the specific meaning of that concept under the constraints of that context. For example (Figure **3-3**), to generate a specific simulation of the concept *near* in the shopping context, the human needs to instantiate the perceptual symbol representing the proximity relation *near* under partial instantiation of the frame representing the concept of the shopping context. So, the simulation of *near* is influenced by instantiation of partial components in the frame representing the context, the shopping event, including the goal of shopping (e. g. for food or clothes), the transportation mode involved in traveling, the traffic situation (e. g. clear or jammed), actual distribution of grocery stores, actual spatial distances between the house location and grocery stores. Therefore, the simulation of a concept is not separable from the context where the concept is used. According to Barsalou (Barsalou, 2000), the context becomes part of the concept structure because of the importance of the meaning of that concept.

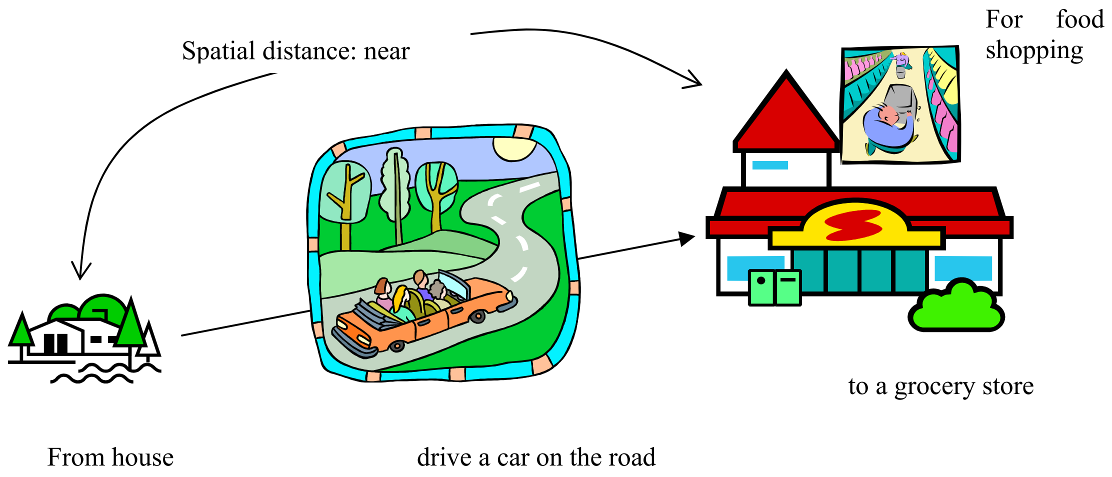
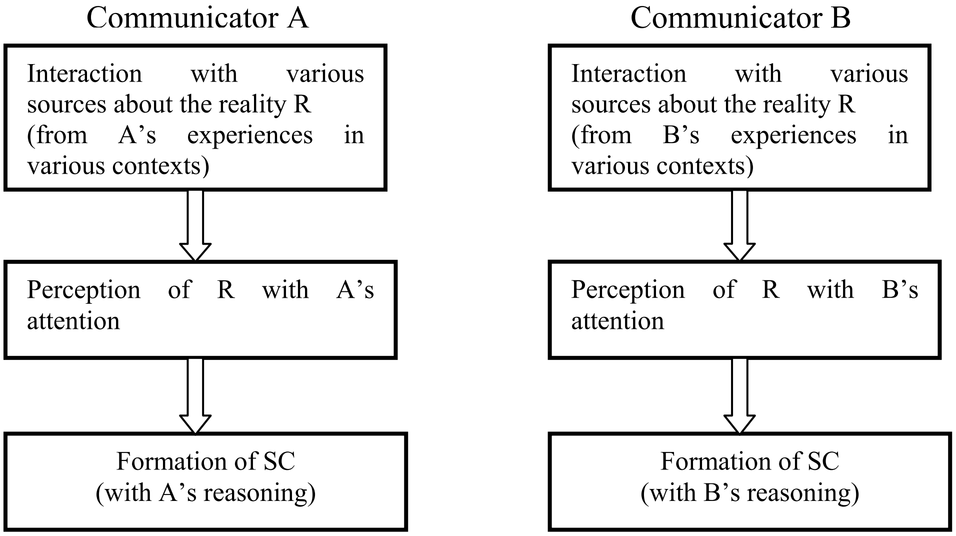


Figure **3-3**: Simulation of the Concept *near*

3.2 Geospatial Knowledge Acquisition

A spatial concept is mentally related to a collection of accumulated knowledge about a category of spatial entities. Therefore, before the human communicator can communicate any spatial concept, the communicator must have formed the meaning of that concept based on knowledge of that concept in memory (Brodeur et al., 2003; Mark et al., 1999). The human's spatial knowledge is usually learned by gradually acquiring elements of the world, e. g. landmarks and routes (Golledge, 1991; Tversky, 1993), and finally integrating them into survey knowledge. According to some researchers (Egenhofer & Mark, 1995; Golledge, 1991, 2002; Kuipers, 1978; Tversky, 1993), there are three major steps involved in the geospatial knowledge acquisition process (Figure **3-4**): 1) interaction with the information resources, 2) perception with attention, and 3) concept formation.



SC denotes a "spatial concept"
R denotes realities that the spatial concept SC refers to

Figure **3-4**: Geospatial Knowledge Acquisition Process

As shown in Figure **3-4**, formation of a spatial concept starts with human interaction with various information sources about the reality (realities) R that the spatial concept refers to. The interaction can be a direct observation of R (Golledge, 1991; Kuipers, 1978; Tversky, 1993) and/or indirect acquisition of preprocessed spatial information, e. g. natural language texts (Taylor & Tversky, 1992), maps (Head, 1991) and other visualization techniques (MacEachren, 1995). The spatial knowledge can be acquired in various contexts (Hirtle & Hudson, 1991; Moar & Carleton, 1982; Presson & Montello, 1988; Thorndyke & Hayes-Roth, 1982). In the example given in Figure **3-4**, two communicators A and B independently form a vague spatial concept from experiencing different realities in different contexts, e. g. for different purposes, in different places, and at different times. Different experiences held by A and B may lead them to form different knowledge as to the same spatial concept.

During the process of perception of the reality R, different people can pay different attention to different aspects of the same reality R, which lead to the "perceptually based biases" during the human perception process (Golledge, 2002). Therefore, even if different people experience the same reality R, they may perceive that reality differently, and form different knowledge about that reality.

In the last step of geospatial knowledge acquisition, the human forms a SPATIAL CONCEPT after reasoning about the perceived information about the reality R. As discussed by Golledge (Golledge, 2002), much of human geospatial knowledge is informal knowledge, and is "acquired using general guidelines that produce vague or error-prone knowledge." People have different reasoning rules for informal geospatial knowledge, which can also lead to different knowledge in different people about the same concept. The “conceptual biases” from “improper thinking and reasoning” (Golledge, 2002) can lead to a knowledge difference on the same spatial concept between different people. It is believed that the geospatial knowledge can also be distorted in the human memory (Lloyd & Heivly, 1987; Tversky, 1981).

3.3 Human-Human Communication Process

With the knowledge of a concept retrieved from the human conceptual system, the human communicators can communicate with each other about that concept. In GIS application domains, there are many examples of interpersonal communication involving spatial concepts, e. g. communication between the GIS operator and the crisis manager (a non-expert user). In such communication, the GIS operator needs to collaborate with the user to reach the user’s spatial information requests involving spatial concepts. Such communication “becomes centered, co-operative, and directed toward some goal” (Cherry, 1978), that is, the communication is around a collaborative discourse.

To facilitate our understanding about human-GIS communication, this section illustrates the human-human communication process with an example of communication between two human communicators, a GIS operator O and a GIS user U, see Figure **3-5** (Schramm, 1954; Tubbs & Moss, 2000). For the convenience of the remaining description about the communication, let O be a male, and U be a female.

In ideal communication, the sender and the receiver should reach the same understanding about the communicated message (Schramm, 1954; Tubbs & Moss, 2000; Weaver & Strausbaugh, 1964). However, real communication never reaches such a success (Heath & Bryant, 2000; Lyotard, 1984; Weaver & Strausbaugh, 1964) and various uncertainties can occur in the communication, e. g. misunderstanding/error (Cherry, 1978; Heath & Bryant, 2000; Weaver & Strausbaugh, 1964), conflict (Heath & Bryant, 2000; Tubbs & Moss, 2000), and vagueness (Dimitrov & Russell, 1994). The human communicators need to handle various uncertainties involved in communication and reach agreement on the meaning of communicated messages.

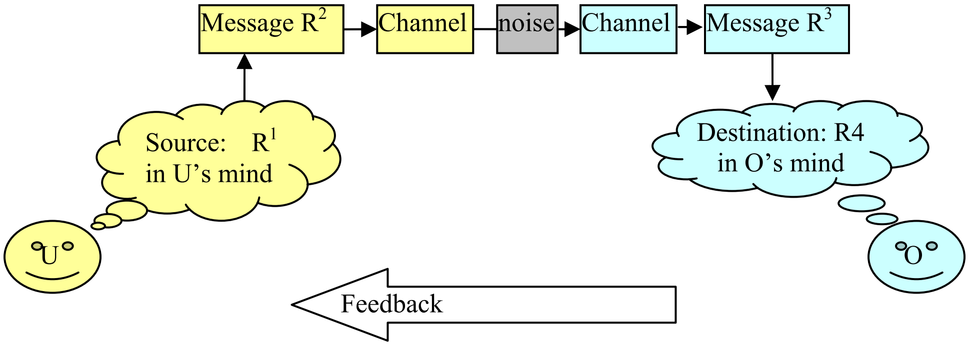


Figure **3-5**: Human Communication Process between a GIS Operator (O) and a user (U)

The remainder of this section describes the human-human communication process in more detail, and then provides an introduction to uncertainty reduction in human-human communication.

3.3.1 Major Steps in Human-Human Communication Process

The communication process between U and O (Figure **3-5**) consists of the following steps:

- 1) The information source R^1 to be communicated is generated in the sender's mind from the sender's conceptual system;

- 2) R^1 is encoded into the message R^2 by the sender;
- 3) The signal representing R^2 is sent out by the sender's communication channels (e. g. the user's sensory organs);
- 4) The signal from the sender with a certain amount of noise is sensed by the receiver's communication channels;
- 5) The signal sensed in the receiver's channels is encoded into message R^3 by the receiver;
- 6) The destination R^4, which represents the receiver's understanding about R^3 corresponding to R^1, is evoked in the receiver's mind from the receiver's conceptual system;
- 7) The receiver starts the feedback process, which is same as the communication process from the sender to the receiver, but in a reverse order.

Steps 1) to 3) represent the encoding process at the sender's side, and steps 4) to 6) represent the decoding process at the receiver's side.

Mental Construction of the Information Source

The information source R^1 (Figure **3-5**) is defined in this study as the mental representation of what the sender tries to communicate to the receiver. R^1 is generated in the sender's conceptual system (Barsalou, 1999). R^1 is a specific simulation of a concept, e. g. a vague spatial concept *near*, that the sender tries to communicate (Barsalou, 1999). So, the information source R^1 is encoded from the sender's conceptual system (Figure **3-6**). The human retrieves the frame/perceptual symbol F^1 of the concept to be communicated and the frame SF^1 of the current context from LTM, and then instantiates F^1 to generate R^1 under the constraints of instantiation of SF^1 (Barsalou, 1999, 2000, 2003).

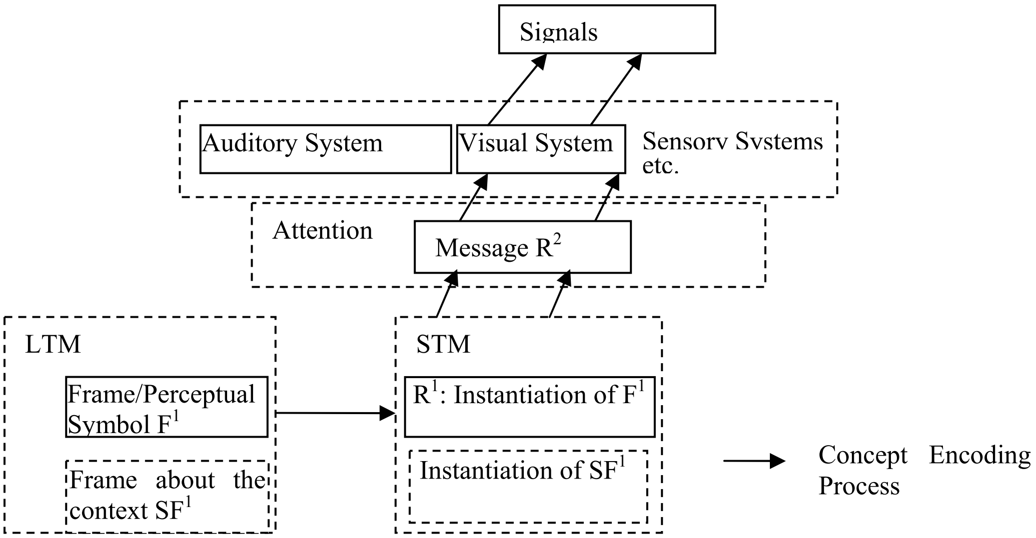


Figure **3-6**: Encoding Process in Human Conceptual System

Encoding Message

To be able to communicate the information source R^1 with the receiver, the sender needs to transform R^1 into a communicable representation of that mental simulation, a message R^2 (Figure **3-6**). The content of R^2 depends on the sender's current attention to R^1 (Figure **3-6**), what representation formats are used and what the sender thinks is enough to represent R^1. Context-dependent properties of the vague spatial concept also contribute to the semantic variability of the message R^2. According to Barsalou (Barsalou, 1982), the contextual-dependent properties of the concept exist in the nature of the concept, but are hard to be encoded in the message R^2. For example, in relation to the vague spatial concept “downtown”, the spatial boundary of “downtown” is context-dependent, and it is hard to be encoded in a verbal message talking about “downtown”.

Sending Signals

The message R^2 (Figure **3-6**) is converted to a signal by the sender’s motor system and sent out. The signal contains what the sender consciously intends to send, and may also contain what the sender unconsciously sends.

Sensing Signal

The first stage in the decoding process at the receiver O’s side is sensing the signal from the sender U by using O’s sensory system (Figure **3-7**). The signal sensed by O may have contained noises from the environment (Figure **3-7**), such as other people’s talking around O and U, which directly affects the receiver’s following perception of the signal.

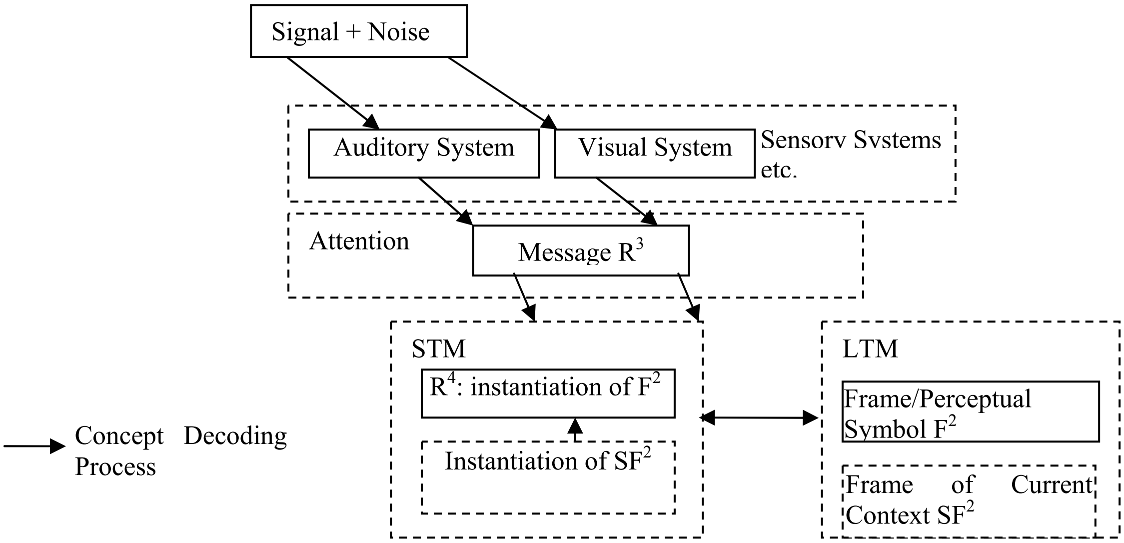


Figure **3-7**: Decoding Process in the Human Conceptual System

Perception of Signal into Message

The second stage in the decoding process is for the receiver to make sense of the signal as a meaningful message R^3 (Figure **3-7**). According to human cognition theories (Lindsay & Norman, 1977; Reisberg, 2001; Wickens, 1984), the receiver's attention at this stage plays an important role in the perception process (Figure **3-7**), and the perception may be selective in terms of what the focus of attention is in the receiver's mind, and what is expected by the receiver. For example, in the case that the receiver O is expecting map-related requests from the sender U, O's perception of the signal would focus on the map-related requests and may ignore U's description about unrelated information, such as what U has eaten.

Meaning Evoked

The last stage of the decoding process is for the receiver O to decode the message R^3 into the information destination R^4 corresponding to R^1 (Figure **3-7**). The message R^3 evokes O to instantiate a perceptual symbol or a frame F^2 as a specific meaning, R^4, of the concept in R^3. Similarly, a super-frame SF^2 representing the current context involving F^2 is instantiated in order to constrain the instantiation of F^2.

Feedback

The communication process does not end until the feedback from the receiver O is received by the original sender U (Tubbs & Moss, 2000; Weaver & Strausbaugh, 1964). In the feedback process, the operator O and the user U reverse their roles in the process from R^1 to R^4, that is, O becomes the communication sender and U is the information receiver. O can return the feedback by showing his understanding about the spatial concept involved in a map representation to satisfy U’s request.

After receiving O’s feedback, U may continue the communication by returning feedback to O until both U and O believe the communication is successful. So, the human-human communication can be a spiral process interleaving conveying messages and feedback between two communicators.

In the communication between the GIS user and the GIS operator, the map representation as the visual feedback helps the user and the operator to understand each other better. This is because the map provides a visual shared spatial context involved in the communication, which can constrain their understandings about the meaning of a spatial concept involved in communication to be closer (MacEachren, 1995; MacEachren et al., 2005).

3.3.2 Strategies of Uncertainty Reduction in Communication

Successful collaborative human communication is to seek consensus between the communicators (Dimitrov & Russell, 1994) through reducing uncertainties involved in communication. The consensus seeking is motivated by the communicators' common concerns, and they agree to explore it together and reach a shared understanding of the common concerns. Three traditional steps are summarized by Dimitrov and Russell (Dimitrov & Russell, 1994) for communicators to build the consensus in communication: 1) the communicators need to set up a "common ground", e. g. find their common interests, goals, and values; 2) they need to exchange their views and opinions and reduce uncertainties by using "an accepted collaborative scheme"; 3) they need to act towards reaching their common concerns.

So, to reach a successful communication, that is, the consensus in communication, human communicators usually start by building their "common ground" (Berger, 1975, 1987; Berger & Calabrese, 1975; Chuah & Roth, 2003; Clark, 1996; Dimitrov & Russell, 1994; Klein et al., 2005), or the shared context (Kang, 1997; Pickering & Garrod, 2004) between the communicators. The shared context constrains the human communicators' understandings about the concept involved in communication. It involves not only the general knowledge of the world and conventions, but also their mutual beliefs on the communication (not only communicative activities, but also other joint activities) history, and the current state of the communication. For the purpose of language comprehension, besides the general knowledge of the world and conventions, the shared context involves at least five dimensions, including time, space, causation, intentionality and protagonists

(Zwaan et al., 1995; Zwaan & Radvansky, 1998). The shared context consists of events involved. Each event is structured with at least these five dimensions.

In case uncertainties exist in communication, the human communicators need to repair the shared context and reduce uncertainties (Dimitrov & Russell, 1994). There are three basic types of strategies for human communicators to reduce uncertainties in communication (Berger, 1987). The first type is passive strategies, that is, communicator A learns about the other communicator B by passively observing B. The second type is active strategies, that is, A asks others about the communicator B whom A is interested in or tries to set up a situation where A can observe B (e.g., taking the same class, sitting close by a table at dinner). Once the situation is set up, A can use either passive or interactive strategies to learn about B. The third type is interactive strategies, that is, A directly communicates with B for further information. In a high information-seeking mode of communication, e. g. interviews, the communicator A can directly ask the other communicator B for questions about B's behavior, goals, and plans.

3.4 Vagueness in Human-Human Communication

This section summarizes origins of vagueness in human-human communication based on the HCF described in Section 3.1, 3.2, and 3.3. For illustration purposes, a simplified chart about information flow involved in communication between a GIS operator O and a user U is provided in Figure **3-8** based on the HCF described above. This figure shows the information flow involved in the communication about the initial request "Show me grocery stores near my house". This section uses this example to explain and summarize the origins of vagueness in human-human communication.

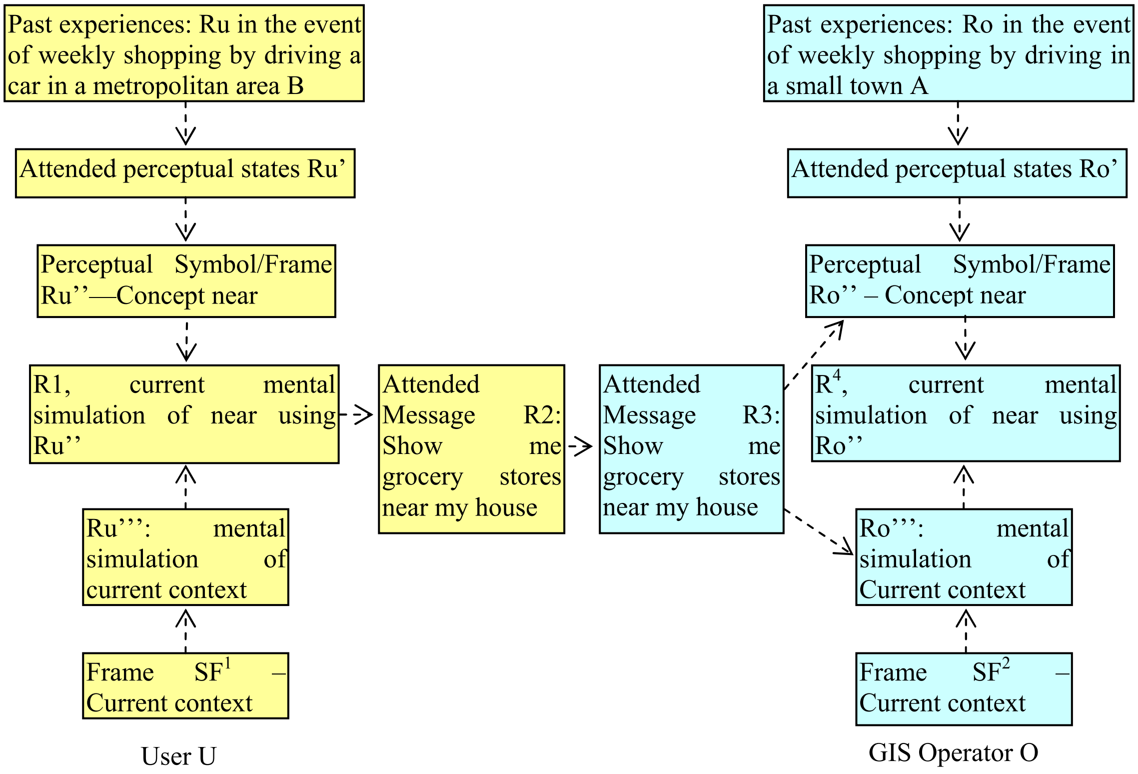


Figure **3-8**: Information Flow in Communication of a Vague Spatial Concept

Practically, the communication of a spatial concept between the GIS operator and the user is considered to be successful when the user is satisfied with the map response showing the meaning of that spatial concept from the GIS operator. Based on the HCF, we can see that the key for a successful communication lies in the two communicators' current mental simulations (that is, current understandings) about the spatial concept involved. In terms of the HFC, the frame/perceptual symbol corresponding to the concept involved is always instantiated under instantiation of its super frame SF representing the current context. Differences between the destination R^4 and the source R^1 in Figure **3-8** directly come from differences between Ru’’ and Ro’’ (individual spatial knowledge), differences between Ru’’’ and Ro’’’ (understanding about the current context), and the inherent fuzziness property of the spatial concept.

The remainder of this section discusses the three major sources contributing to the vagueness problem in human-human communication: 1) individual differences on spatial knowledge acquisition, 2) different understandings about the current context, and 3) the inherent fuzziness of the vague spatial concept.

Individual Differences on Spatial Knowledge Acquisition

The individual abstract concept knowledge difference between Ru’’ and Ro’’ corresponding to the spatial concept communicated (Figure **3-8**) comes from individual knowledge acquisition process differences. Firstly, different past experiences between Ru and Ro can lead to the development of different perceptual symbols on the same concept. In the example of communication of *near* shown in Figure **3-8**, U develops perceptual symbols of *near* corresponding to less than half an hour of driving distance from her past experience of weekly food shopping in a metropolitan area. O forms perceptual symbols of *near* corresponding to less than 10 minutes of driving distance from his past experience of weekly food shopping in a small town. Secondly, even if U and O develop perceptual symbols of *near* from same the shopping experiences, their different levels of attention paid can lead to different perceptual states Ru’ and Ro’ perceived by them. Thirdly, different reasoning involved in transforming perceptual states into perceptual symbols can lead to different results of perceptual symbols formed.

Different Understandings about the Current Context

In the example of communication of *near* (Figure **3-8**), mental simulations of the current context, Ro''' and Ru''' come from instantiations of the frames representing the current context SF^1 and SF^2, respectively. Differences between Ro''' and Ru''' can come from different instantiations of SF^1 and SF^2. SF^1 and SF^2 may represent different concepts/contexts. In case SF^1 and SF^2 refer to the same context, U and O may instantiate them with different values of sub-regions, and/or in different depth (due to recursive structures of the frames) and/or with different width (due to different attentions paid to the same context).

Inherent Fuzziness

Perceptual symbols corresponding to the same concept are associated together with that concept. The inherent fuzziness is encountered during the categorization process for the human to classify a perceptual symbol into the association. For example, in the context of Bob's weekly food shopping in a metropolitan area, Bob can easily categorize perceptual symbols referring to less than 10 minutes of driving distance into the concept *near*, and perceptual symbols referring to greater than one hour of driving distance into the concept *far*. However, it is hard for him to categorize perceptual symbols referring to spatial distances in between. The inherent fuzziness during the categorization process can lead to the inherent fuzziness of the mental simulation of the concept, R^1 and R^4.

3.5 Human Communication Principles

Based on the HCF described at the beginning of this chapter, this section summarizes human communication principles (HCPs), that is, human solutions, used to handle the vagueness problem. Here, we focus on those related to understanding the meaning of a vague spatial concept (HCP1), and those related to how to reduce vagueness involved in communication (HCP2).

HCPs for Understanding about the Meaning of a Vague Spatial Concept

Here, we summarize HCP1 as follows:

- a) The human understands the meaning of a vague spatial concept under constraints of mental simulation of the current context.
- b) The human usually attempts to collect contextual knowledge before interpreting the meaning of the vague spatial concept.
- c) The human usually has only partial contextual knowledge. This means that the human communicator usually interprets the meaning of a vague spatial concept with major context information only, not all information about the current context. The communicator's model of the meaning of the vague spatial concept under constraints of a series of major context factors can be considered as a function f_A of a list of variables (see Eq. **3.1**):

$$M_A = f_A(T_1, T_{2,} ..., T_i, ..., S_1, S_2, ..., S_i, ..., H_1, H_{2,} ..., H_i, ...) \qquad \textbf{3.1}$$

where: A denotes the communicator agent A; T_i denotes the *i*th task context factor; S_i denotes the *i*th spatial context factor; H_i denotes the *i*th human context factor.

HCPs for Handling the Vagueness Problem

HCP2 are summarized as follows:

- a) The human communicators always start by building a shared context first, and further reduce vagueness later, in order to handle the vagueness problem involved in communication.
- b) The shared context in communication involves five dimensions, including time, space, causation, intentionality, and protagonist/objects involved in communication.
- c) The human communicators can use various collaborative dialogue strategies to repair and improve the shared context. Here, we define the interactive strategies (Berger, 1987) for direct communication of context information as the collaborative context-sharing dialogue strategy. For example, with this strategy, the GIS operator can actively ask the user for the user's task information or actively show certain spatial information useful to the user without being asked.
- d) After the shared context is built and repaired, if needed, the communicators can ground their understandings to a common understanding through various collaborative dialogue strategies. Here, we

define the interactive strategies for refining the meaning of the vague spatial concept based on its previous meaning as the negotiation strategy (Dimitrov & Russell, 1994; Kang, 1997). For instance, in communication about *near* in the food shopping context, with this strategy, the GIS operator can ask the user's opinion about increasing or decreasing the *near* distance shown on the map, and the user can also directly give such opinions without being asked. We also define the interactive strategies for direct communication of the meaning of the vague spatial concept as the direct communication strategy. For example, in communication about *near* in the food shopping context, the user can directly tell the GIS operator or point out which grocery stores he/she believes are *near*, and the GIS operator can also ask for the user's understanding about *near*.

3.6 Summary and Discussion

This chapter describes the major components of the HCF in detail. The primary origins of the vagueness problem and human communication principles involved in human-human communication are analyzed based on the HCF.

The HCF involves three major components, including the spatial knowledge acquisition process, the human conceptual system, and the human-human communication process. The spatial concept as human spatial knowledge is acquired through a human's experiences, perception, and reasoning about spatial information sources. Human's spatial knowledge is stored and accumulated as perceptual symbols/frames in LTM of the human conceptual system. In the STM of the human conceptual system, the meaning of a vague spatial concept is generated as a specific instantiation of the perceptual symbol/frame corresponding to that concept under constraints of mental simulation of the current context. With the spatial knowledge acquired, the human conceptual system generates the information source from the speaker's side and the information destination invoked by the receiver in human-human communication. In collaborative human-human communication, the human communicators seek consensus, that is, the success of the communication, through reducing vagueness in various collaborative ways.

The HCF shows that vagueness in human-human communication involves three origins, including individual spatial knowledge differences, different understandings about the current context, and the inherent fuzziness. We can see that the context-dependency problem, as the focus problem in this study, is caused by individual differences in spatial knowledge and by different understandings about the current context. The HCF also shows HCPs that human communicators use to understand the meaning of a vague spatial concept and handle the vagueness problem in communication. In high-information seeking communication, e. g. face-face communication, the human communicators usually start by building the shared context, and further reduce vagueness with various interactive collaborative strategies.

The HCF provides a comprehensive explanation about the vagueness involved in human communication of spatial concepts, which facilitates our understanding about the vagueness involved in human-GIS communication. It also shows success and effectiveness of human collaborative dialogues for human communicators to handle the vagueness problem. Based on the assumption that a better match with the human cognition will lead to a better GIS interface, we believe that these human collaborative dialogues and associated HCPs would direct a better approach to handle the vagueness problem involved in human-GIS communication.

Chapter 4

Methodology

The success and effectiveness of human collaborative dialogues indicate the need for collaborative human-GIS dialogues. This need drives us to propose and develop a new approach, a collaborative dialogue approach, for the GIS to handle the context-dependency problem. An agent-based computational model, the *PlanGraph* model, is developed to support this approach. This chapter describes the collaborative dialogue approach and the *PlanGraph* model in detail.

The remaining chapter is organized as follows. Firstly, an overview about this chapter is provided in Section 4.1. Section 4.2 explains our understanding of the context-dependency problem, in human-GIS communication based on the HCF. Section 4.3 describes the human emulation principle of the collaborative dialogue approach. Details about the *PlanGraph* model are given in Section 4.4. An example of human-GIS collaborative dialogue is explained in Section 4.5. This example illustrates how the collaborative dialogue approach and the *PlanGraph* model can help the GIS and the user to handle the context-dependency problem. Summary and discussion conclude this chapter.

4.1 Overview

As discussed in Chapter 3, the HCF is developed to facilitate our deeper understanding and better handling of the vagueness problem in human-GIS communication. Based on the HCF, we have a better and more comprehensive understanding about the vagueness problem in human-GIS communication, e. g. how the vagueness problem in particular, the context-dependency problem, is generated in human-GIS communication, what contributes to vagueness in human-GIS communication, why the shared context is needed between the GIS and the user, why a collaborative human-GIS dialogue is needed for handling vagueness in human-GIS communication, and various types of the contextual factors involved in the shared context.

The success and effectiveness of human-human collaborative dialogues and the distributed nature of context-knowledge in human-GIS communication lead us to consider developing a collaborative dialogue approach for the GIS to handle the context-dependency problem, that is, enabling human-GIS collaborative dialogues for the GIS and the user to build a shared context and further reduce vagueness involved in communication. The major design principle of the collaborative dialogue approach is to enable the GIS to handle the context-dependency problem by emulating the human communicator's collaborative solutions.

The agent-based computational model, the *PlanGraph* model, is developed for implementation of the collaborative dialogue approach. This model is designed based on agent-based theories and models, and further extended by following the HCF, in particular, HCPs involved in handling the vagueness problem. This model includes two major components, the *PlanGraph* and associated reasoning algorithms. The *PlanGraph* is a complex data structure which represents the GIS agent's knowledge on the dynamic human-GIS collaboration process underlying the human-GIS collaborative communication. The reasoning algorithms associated with the *PlanGraph* are designed to support the GIS agent to reason about how to collaborate with the user, in particular, how to handle the context-dependency problem through collaborative human-GIS dialogues.

The GIS agent with the collaborative dialogue approach and the *PlanGraph* model can collaborate with the user toward the success of the user's request. The system can keep track of the shared context between the GIS and the user by utilizing of the *PlanGraph*. In case the shared context needs to be improved, the system can repair it through various context-sharing human-GIS dialogues. The system can further have a shared or closer understanding about a vague spatial concept involved in communication with the user under constraints of the shared context represented in the *PlanGraph*. In case vagueness still exists, the system can further reduce it through various collaborative human-GIS dialogues by following human solutions in this case. Finally, the system and the user can reach agreement on the meaning of the vague spatial concept.

4.2 Vagueness in Human-GIS Communication

By referring to the HCF, we can understand how the GIS and the human user communicate a vague spatial concept (Figure **4-1**), and how the vagueness problem is generated in human-GIS communication.

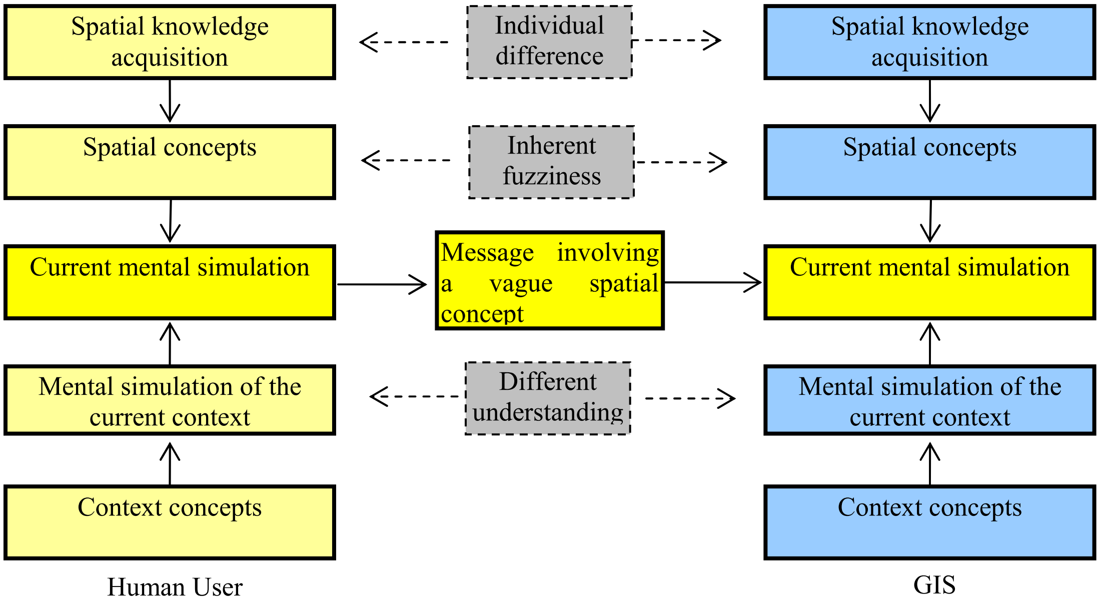


Figure **4-1**: Human-GIS Communication of Vague Spatial Concepts

Origins of Vagueness in Human-GIS Communication

In human-GIS communication of a vague spatial concept (Figure **4-1**), both the GIS and the human user understand the meaning of a vague spatial concept based on their different abstract spatial conceptual knowledge under constraints of their different understandings about the current context. Usually, the abstract spatial concept knowledge built in GIS comes from certain human subjects' spatial knowledge. Therefore, like vagueness in human-human communication, vagueness in human-GIS communication also originates from three aspects, including individual differences in spatial knowledge acquisition, differences in understandings of the current context involved in communication, and the inherent fuzziness in vague spatial concepts. Similarly, the context-dependency problem in human-GIS communication also comes from different individual abstract spatial concept knowledge and different understandings about the current context between the GIS and the user.

Various Types of Contextual Factors in Human-GIS Communication

The current context behind the user request to the GIS influences the meaning of a vague spatial concept involved in the request. Based on the five dimensions of the shared context in human-human communication (see HCP2(b)), we can also classify the contextual factors in human-GIS communication into five different types, including the temporal context (temporal contextual factors), the spatial context (spatial contextual factors), the causal context (causal contextual factors), the task context (intentionality related contextual factors), and the human context (human subject related contextual factors). These various types of contextual factors are integrated around an event behind the user request. In this study, the *task context*, the *spatial context* and the *human context* are considered as three major types of contextual factors that influence the meaning of a vague spatial concept (Cai et al., 2003), based on previous studies reviewed in Chapter 2 that discuss the context dependency aspect of the vague spatial concept or handle the context-dependency problem in human-GIS communication.

Distributed Context Knowledge in Human-GIS Communication

Human-GIS communication usually starts with the human user's spatial information request to the GIS. At the beginning of human-GIS communication, information about the context behind the user's request is distributed between two ends of the communication. When the spatial information request is communicated, concerning the current context behind the user's request, the user usually holds more knowledge about the task context and the human context than the GIS, and the GIS usually holds more knowledge about the spatial context through the GIS spatial database (Figure **4-2**). For example, as the shopping context behind the user's spatial information request, "I would like to see the map of grocery stores near my house", the user is more knowledgeable about the task for which the map is used (e. g. the user's shopping goal, the transportation mode involved in traveling between the house and grocery stores, etc.) and the user's own subjectivity related to his/her abstract spatial concept knowledge (e. g. the user's shopping experiences, preferences on understandings about the vague spatial concept, *near*, in different contexts, etc.); while the GIS may be more knowledgeable about details of the spatial area (e. g. the spatial scale/size of the area considered, details of the traveling routes, absolute and relative spatial distances between the user's house and grocery stores). Therefore, at the beginning of the communication, knowledge about the current context behind the user's spatial information request is distributed between the user and the GIS.

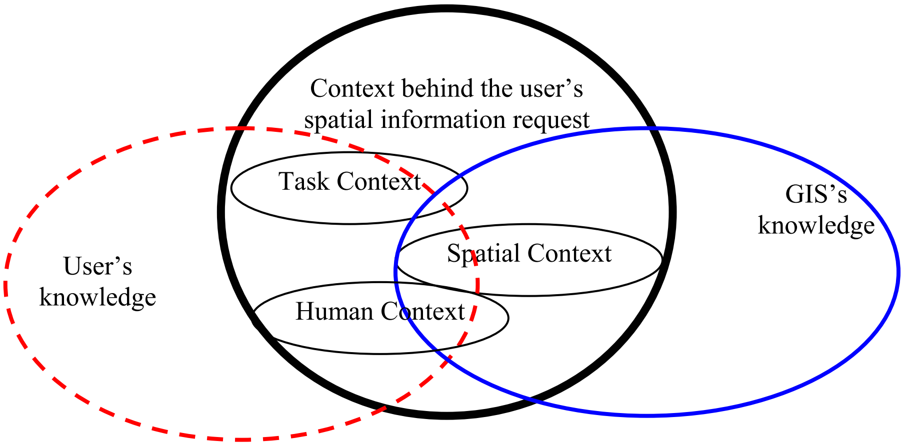


Figure **4-2**: Distributed Context Knowledge between GIS and User

Need for Building a Shared Context

Due to the distributed nature of the context knowledge at the beginning of human-GIS communication, the GIS and the user also need to build a shared context (that is, a shared understanding about the current context behind the user's spatial information request) to reduce vagueness involved in communication. This is because, similar to human-human communication, the shared context in human-GIS communication can constrain the GIS and the user's understandings about the same vague spatial concept.

Need for Collaborative Human-GIS Dialogues

In order to build a shared context in human-GIS communication of a vague spatial concept, the spoken language enabled GIS can follow the way that human communicators take to build a shared context. As described in Chapter 3, one effective way for human communicators to build the shared context is through varieties of context-sharing dialogues (see HCP2(c)). Thus, collaborative context-sharing dialogues between humans and GIS would help the two agents to build a shared context.

Similar to the human-human communication, vagueness in human-GIS communication may still exist after the shared context is built. To further reduce vagueness, various collaborative human-GIS dialogues are also needed so that the system and the user finally reach a shared understanding about the vague spatial concept.

4.3 Design Principle of the Collaborative Dialogue Approach

We propose and develop a collaborative dialogue approach for the GIS to handle the context-dependency problem through collaborative human-GIS dialogues, by considering the need for the GIS to build a shared context and further reduce vagueness through collaborative human-GIS dialogues. The major design principle of this approach is the human emulation principle, that is, to enable the GIS to emulate a human agent's role to handle the context-dependency problem through collaborative human-GIS dialogues. Considering that the system needs to help the user for his/her spatial information needs, we use the human GIS operator as the human agent that the GIS emulates in human-GIS communication.

Under the human emulation principle, human-GIS communication is modeled as a human-GIS collaborative planning process toward the common goal between the GIS and the user, which is the success of the user's spatial information request. Here, the assumption is that both the system and the user are rational agents (Bratman, 1992). The system and the user both should contribute to the human-GIS collaboration toward the success of the user's request. The user can contribute more to proposing his/her request, that is, the common goal of the collaboration, and evaluating the map result from the GIS. He/she also needs to help the system if needed, such as answering the system's question. The system can contribute more to coming up with an executable plan toward the success of the goal, executing GIS command related actions and returning the map and/or text/speech responses to the user. The system can also ask the user for help when needed. During the human-GIS communication process, both the system and the user can contribute to building the shared context. With collaborative dialogues, the system and the user can exchange relevant context information so that part of individual context knowledge can become shared and integrated into a consistent set of knowledge held by both participants. As a consequence, the system and the user can finally reach a shared understanding about the context involved in communication. Under the constraints of the shared context built in communication, the GIS and the user can reach a shared/closer understanding about the meaning of the vague spatial concept initially involved at first. The meaning of the vague spatial concept can be further grounded into a common understanding through various human-GIS collaborative dialogues, e. g. the negotiation strategy or the direct communication strategy, if needed.

One of the keys for the GIS with the collaborative dialogue approach is to keep track of the current state of the human-GIS collaboration underlying the human-GIS communication, that is, the shared context between the system and the user for the two agents to understand each other in communication. By following the shared context built in human-human communication, the shared context in human-GIS communication consists of general knowledge involved in human-GIS collaboration and its current state in communication, and the communicators' mental states kept on the communication. The general knowledge involved in human-GIS collaboration is similar to the human's abstract concept knowledge in long term memory. The current state of such general knowledge in collaboration is similar to the human's mental simulation of these concepts in short term memory.

The GIS should keep track of various contextual factors involved in communication. The collaborative dialogue approach focuses on modeling the three major types of contextual factors to the vague spatial concept, including the task context, the spatial context, and the human context. The three types of contextual factors are organized in terms of the knowledge structure of the event involving the user's spatial information request. The task contextual factors include the user's final goal, and actions and associated facilities involved in the event. The spatial contextual factors include spatial locations and their associated spatial attributes where the user's actions take place. The human context includes the human user involved in human-GIS communication and the underlying human-GIS collaboration.

The GIS with the collaborative dialogue approach needs to have reasoning capabilities similar to the human's reasoning capabilities, in order to emulate the human GIS operator's role to reduce vagueness. The reasoning algorithms in the collaborative dialogue approach are designed for the GIS to reason about how to collaborate with the user toward the success of the user's spatial information request, in particular, how to handle the context-dependency problem involved in communication.

4.4 *PlanGraph* Model

The *PlanGraph* model is designed and developed for the GIS to handle the context-dependency problem through the collaborative dialogue approach. It helps to keep track of the shared context in the human-GIS communication. It also provides automatic reasoning algorithms for the GIS to reason about how to collaborate with the user, in particular, about how to build a shared context with the user and further reduce vagueness in communication through various collaborative dialogues.

4.4.1 Computational Theoretical Framework

We apply the agent-based computational technology to support the collaborative dialogue approach, considering the need to enable the system to have comparable capabilities with respect to a human GIS operator. This is because the agent-based computational technology fits with the human emulation principle of this approach and supports more flexible and complex communication between the system and the user (McTear, 2002). For comparison between the agent-based technology and other technologies used for human-computer dialogue management, see the review paper by McTear (McTear, 2002).

The *PlanGraph* model is mainly designed based on two important computational theories, the computational collaborative discourse theory (Grosz & Sidner, 1986) and the *SharedPlan* theory (Grosz & Kraus, 1996, 1999), and the agent-based computational model, the *RecipeGraph* (*Rgraph*) model (Lochbaum, 1994, 1998). This model extends the *Rgraph* model by following the computational discourse theory and the *SharedPlan* theory. It is further extended to facilitate the GIS agent to handle the context-dependency problem by following the HCF, in particular, by following the HCPs for human communicators to handle the context-dependency problem.

The assumption under our design of the *PlanGraph* model is that collaborative behavior between the GIS and the user underlying the collaborative human-GIS discourse is centered on a collaborative plan contributed to by both the system and the user. The system agent needs to retain knowledge on the collaborative plan so that it can reason about how to help the user in communication. There are several plan-based computational theories, e. g. *SharedPlan* (Grosz & Kraus, 1993, 1996), *Joint Intention* (Cohen & Levesque, 1991), and *Belief-Desire-Intention* (*BDI)* model (Bratman, 1987). The *SharedPlan* theory was developed mainly based on the computational collaborative discourse theory (Grosz & Sidner, 1986) to model the collaborative plan of a group of agents in collaboration. Therefore, we believe that it can fit our need to model the collaborative plan between the GIS and the user underlying collaborative human-GIS communication. According to the *SharedPlan* theory, a *SharedPlan* involves a set of mental states (individual/group intentions, individual and mutual beliefs and commitment) that participating agents in collaboration hold on actions involved in the collaboration. It is these mental states that lead the agents to take collaborative actions to help each other during the collaboration.

The *PlanGraph* model needs to enable the GIS to understand and manage the collaborative human-GIS communication, in particular, the intentions involved in human-GIS collaboration underlying the communication. There are two major approaches for understanding the collaborative discourse structure, including the informational approach and the intentional approach (Lochbaum et al., 2000). We believe that the intention underlying the user's utterances is the major factor that influences the collaborative behavior, e.g. actions, questions, and answers, that the user and the system will take in communication. Therefore, we adopt the intentional approach in this study. The computational collaborative discourse theory developed by (Grosz & Sidner, 1986) plays an influential impact on studies with the intentional approach, and it is the base of the *SharedPlan* theory. Thus, we design the *PlanGraph* model based on this computational collaborative discourse theory to facilitate the system's understanding about human-GIS communication. According to the computational discourse theory, the structure of a collaborative discourse consists of three interrelated structures, a linguistic structure, an intentional structure and an attentional state. The linguistic structure consists of discourse segments (DSs), a sequence of utterances, and relations embedded among the DSs. There is always a purpose underlying each DS (DSP). The DSP functionally contributes to the achievement of higher level intentions underlying the higher level DSs. The intentional structure consists of the DSPs and relations among the DSPs. An attentional state represents the participants' focus of attention at a moment during the collaborative discourse. The attentional state always shifts, depending on whether the current focus intention/DSP has been fulfilled or partially fulfilled.

The *PlanGraph* model is designed to keep track of the current state of the human-GIS collaboration and provide reasoning algorithms for the GIS, by extending the *Rgraph* model. The *Rgraph* model is designed based on the computational discourse theory and the *SharedPlan* theory. The *Rgraph* represents mental states that the agent being modeled (e. g. the GIS agent in this study) holds about how all actions for intentions underlying the discourse are related at a given point in the discourse. It is used mainly for modeling the intentional structure of the collaborative discourse. The *Rgraph* model provides reasoning algorithms to interpret the contribution of each utterance in the discourse to the current *SharedPlan* underlying the discourse. Thus, we extend the *Rgraph* model for the *PlanGraph* model in order for the GIS to keep track of the *SharedPlan* between the system and the user and reason about how to handle the vagueness problem in collaborative human-GIS communication.

4.4.2 Basic Concepts

We define several basic concepts involved in the *PlanGraph* model before describing this model in detail. These concepts include *action*, *recipe*, and *plan*.

Action

An *action* refers to an act to be performed to fulfill a certain goal. An action can be either a basic action or a complex one. A basic action defined in this study is an action that can be executed directly. “Display a named layer” and “Refresh a map” are two examples of basic actions. A complex action is defined in this study as an action that can not be directly executed, such as an action which contains several subactions, an action that requires a set of knowledge-preconditions to be satisfied before it can be executed, and an action which the agent has multiple possible ways to do.

Recipe

The agent has one or multiple possible ways to execute a complex action α. One possible way is one recipe of α. A recipe for a system action α encodes the system’s knowledge about the abstract structure of that action, that is, how to do that action. It is a one-level schematic description of the system’s knowledge, usually, in terms of parameters, subactions, and constraints necessary for executing that action α (Figure **4-3**).

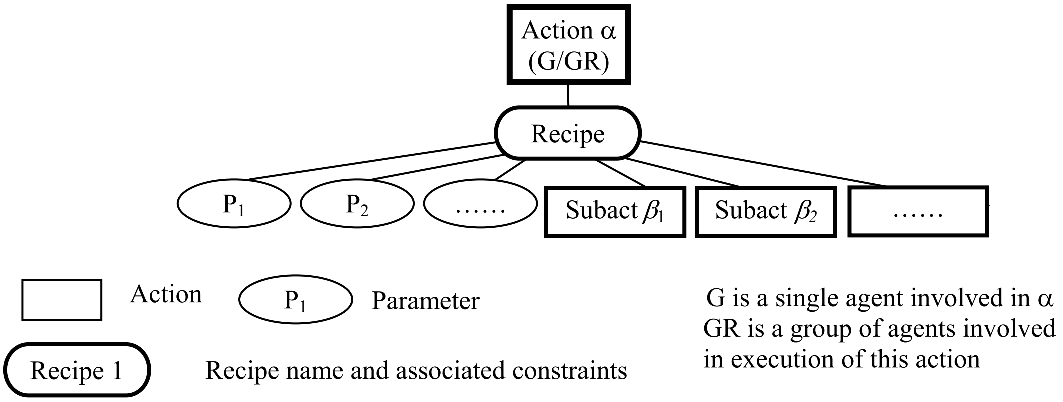


Figure **4-3**: Structure of a Recipe

The constraints defined in a recipe can be any pre-conditions and/or time orders of instantiating parameters and executing the subactions. For example, in Figure **4-3**, parameters P_1 and P_2 need to be instantiated before execution of the subactions (β_1 and β_2), and β_1 needs to be executed before execution of β_2. Currently, the parallel execution of actions or instantiation of parameters are not considered in this study because in most cases the GIS needs to instantiate parameters and do actions one by one. Parameters are modeled as knowledge pre-conditions for the recipes. A parameter in a recipe of an action can either be required or optional in this study. If it is required, the predicating agent of that action must instantiate it before executing the sub-actions in that action. If the parameter is optional, the participating agent can instantiate it by estimation if it can not be instantiated from the user's request or the system's spatial database. Subactions in the recipe have access to all the parameters in the recipe, and cannot be executed before required parameters of the recipe are instantiated and constraints associated with the recipe are satisfied.

Plan

A *Plan* for an action α specifies who is going to do α and the agent's mental states on α. α can be done by a single agent or multiple agents. The plan to be done by a single agent is an individual plan, and the plan to be done by a group of agents is a *SharedPlan*.

An individual agent G must have the following mental states on α (Grosz & Kraus, 1996) to have a full plan to do an action α: (1) G has a recipe R for α; (2) for each constituent act β_i of R, G intends to perform β_i; G believes that it can perform β_i; G has an individual plan for β_i (Lochbaum, 1998).

A *SharedPlan* is constructed by the agents in the collaboration. It encodes mental state specifications that a group of agents hold on the plan for an action α (Grosz & Kraus, 1996, 1999; Grosz & Sidner, 1990; Lochbaum, 1994). It consists of subplans, which can be individual *plans* and/or *SharedPlans*. A group of agents (GR) must hold the following mental states on α to have a full *SharedPlan* for α:

- GR is committed to performing α and has a recipe to do α.
- For each single-agent constituent act β_i of the recipe, there is an agent $G_{\beta i}$ (belonging to GR), who has an individual plan for β_i, and GR mutually believe in $G_{\beta i}$'s capabilities for doing β_i, and are committed to $G_{\beta i}$'s success in $\beta_{i.}$
- For each multi-agent constituent act β_i of the recipe, there is a subgroup of agents $GR_{\beta i}$ (belonging to GR), which has a *SharedPlan* for β_i, mutually

believe they can do β_i, GR mutually believe $GR_{\beta i}$'s capabilities for doing β_i, and are committed to $G_{\beta i}$'s success in β_i.

A *SharedPlan* is full when all the required mental states have been established, otherwise it is partial. The collaboration process for α is a plan evolution process of a *SharedPlan* toward the success of α from partial to complete (Grosz & Kraus, 1999).

4.4.3 Structure of the *PlanGraph* model

The *PlanGraph* model is designed to support application of the collaborative dialogue approach in human-GIS communication of vague spatial concepts. This model enables the agent being modeled (that is, the GIS agent, in this study) to keep track of the current state of human-computer collaboration through use of *PlanGraph* (Figure **4-4**). It also provides a set of reasoning algorithms associated with the *PlanGraph*. With these reasoning algorithms, the GIS agent can reason about how to process and manage human-GIS communication/collaboration with the use of the *PlanGraph*.

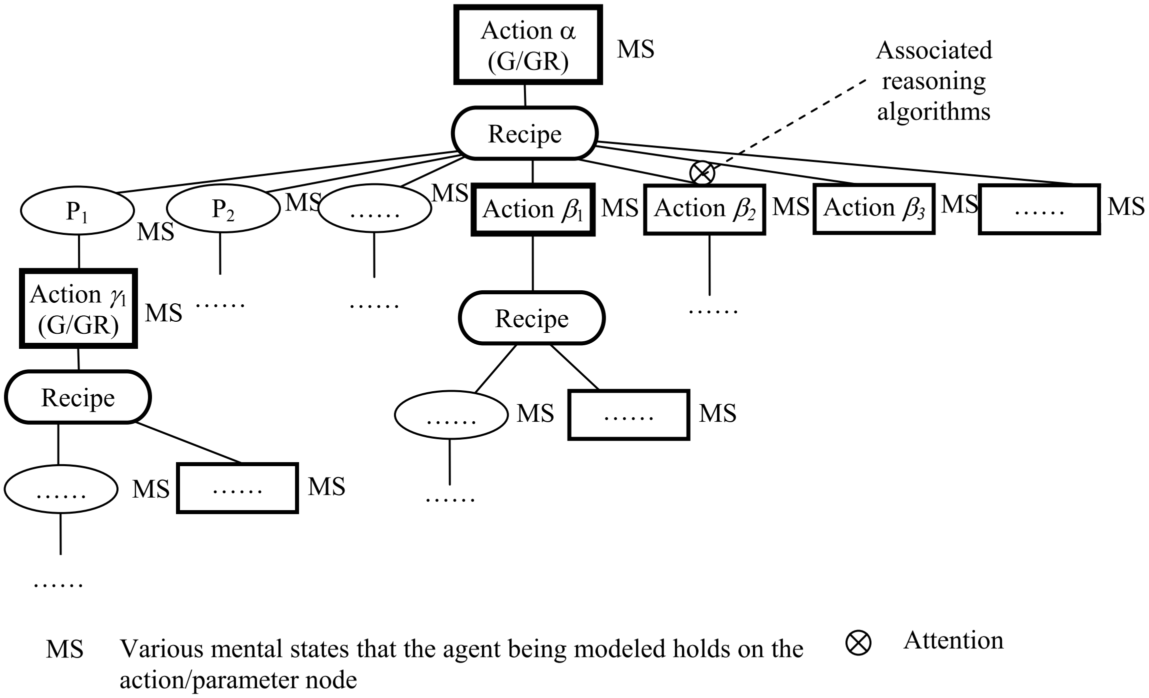


Figure **4-4**: Structure of the *PlanGraph* model

The *PlanGraph* represents the dynamic knowledge that, at a given time point during the human-computer collaboration, the agent being modeled (that is, the GIS agent, in this study) adheres to the *SharedPlan* developed by the agents in collaboration during the collaborative planning process. It consists of the root plan representing the common goal of the agents in collaboration, and all its sub-plans (Figure **4-4**). At that given time point, part of the subplans in the *SharedPlan* may have been executed, e. g. the plan for action γ_1 and the plan for action β_1 in Figure **4-4**. Part of them may be on the attention focus for elaborating and executing, e. g. the plan for action β_2 in Figure **4-4**. Part of them may be waiting for execution toward the success of the common goal, e. g. the plan for action β_3 in Figure **4-4**.

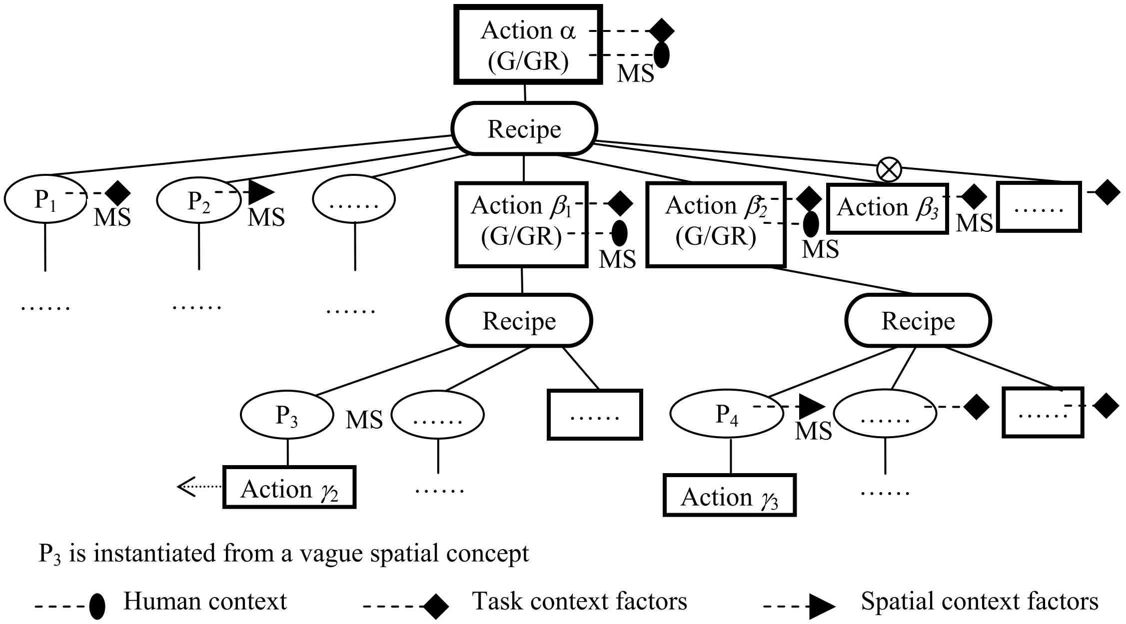
Each node in the *PlanGraph* is centered around a plan/subplan, a complex data structure which records an action (and associated parameters if possible) together with a set of mental states (MS in Figure **4-4**) that the agent being modeled holds on that action. A plan/subplan about how to reach an action involved in the *SharedPlan* comes from the knowledge of the agent(s) responsible for that action. During the collaborative planning process, this agent(s) needs to decide an appropriate recipe for that action from its best knowledge and have it executed. The GIS agent must have general knowledge of such actions and recipes involved in human-GIS collaboration in order to construct the *PlanGraph* during the human-GIS collaborative planning process.

The *PlanGraph* records the agent(s)'s mental states on not only actions, but also parameters. The set of mental states on an action/parameter consists of the intention, commitment, and mental beliefs that the agent(s) responsible for that action/instantiation of that parameter holds on that action/parameter.

The *PlanGraph* also records the attention focus that the agent being modeled holds on the *SharedPlan* built in the collaboration at a given point of the collaborative planning process. During the collaboration process, the agent being modeled can reason about when to shift the attention focus on the *SharedPlan* from one node to another, depending on whether the current focus intention on an action/parameter node has been fulfilled or not.

4.4.4 Modeling Various Contextual Factors

By modeling the dynamic human-GIS collaborative planning process, the *PlanGraph* can help the system to keep track of various contextual factors involved in the shared context between the system and the user. Such contextual factors can be modeled as different components of the *PlanGraph* (Cai et al., 2003; Cai & Xue, 2006), depending on how they contribute to the human-GIS collaboration (Figure **4-5**). As discussed in Section 4.1, there are three major types of contextual factors which influence the meaning of a vague spatial concept in communication. The meaning of the concept is usually used by the system to instantiate a parameter, e. g. P_3 in Figure **4-5**, involved in a GIS command. The remaining section describes how these three major types of contextual factors are modeled in *PlanGraph*.



P_3 is instantiated from a vague spatial concept

Figure **4-5**: Contextual Factors Modeled in *PlanGraph*

Human Context as Agent(s) of Plans

The agent(s) (e. g. G/GR in Figure **4-5**) responsible for a plan/subplan in the *SharedPlan* built by the agents in collaboration is considered as the human context influencing the meaning of a vague spatial concept. With the human agent modeled in the *PlanGraph*, during the collaborative human-GIS communication, the GIS agent can determine the human context from the *PlanGraph* to constrain its understanding about a vague spatial concept. For example, from the *PlanGraph* representing the *SharedPlan* between the GIS and the user, the system knows who proposes the spatial information request involving a vague spatial concept. As a result, it can retrieve more human context information (e. g. gender, age, familiarity with the spatial context involved, ethnicity, etc.) about this particular user from its knowledge base and use the retrieved human context information to constrain its understanding about the vague spatial concept.

Task Context as Actions and Parameters of Plans

The task context in the human-GIS communication usually involves the user's goals/activities and facilities associated with these goals/actions. In the *PlanGraph*, the user's goals/activities are modeled as action nodes, and facilities used in activities for these goals are modeled as parameters associated with these action nodes. For example, P_3 in Figure **4-5** is instantiated from the meaning of a vague spatial concept, e. g. *near*. The goals/activities, e. g. the food shopping purpose and the traveling activity, involved in the task context are modeled as ascendant action nodes (e. g. action α, and β_1 in Figure **4-5**) of P_3 and their descendant action nodes (e. g. action β_2 and β_3, and their descendant nodes in Figure **4-5**). Facilities used in these activities, e. g. the transportation vehicle, are modeled as part of parameter nodes associated with these action nodes, e. g. the parameter node P_1 in Figure **4-5**.

With task context factors modeled in the *PlanGraph*, during the collaborative human-GIS communication, the GIS agent can try to build a shared understanding with the user about the task context factors that influence the meaning of a vague spatial concept. The shared task context information will further help the GIS and the user to reach closer understandings about the meaning of a vague spatial concept.

Spatial Context as Parameters of Plans

The spatial context in the human-GIS communication usually involves the spatial properties of the spaces where the user's activities towards the user's goal behind the user's spatial information request take place. These spatial properties, e. g. spatial area and routes involved in the user's traveling activity, usually are modeled as part of parameters associated with the action nodes representing these user activities. For example, actions α, β_1, β_2, and β_3 in Figure **4-5** represent the user activities toward the user's goal in the current context, and the spatial context factors can be modeled as part of parameter nodes associated with these action nodes. P_2 and P_4 are two example parameter nodes representing the spatial context factors.

With the spatial contextual factors modeled as parameter nodes in the *PlanGraph*, the GIS agent can determine which spatial context information has been shared/not shared with the user, and then decide if it needs to communicate with the user for reaching a shared understanding about the spatial context involving the vague spatial concept. Because the GIS map is usually used in human-GIS communication, all of the spatial properties of the spaces can be shown on the map. So the spatial context information shown on the GIS map can directly facilitate the GIS and the user in reaching a shared understanding about the spatial context involved in communication (MacEachren & Cai, 2006)

In summary, the GIS can keep track of the current context involving the vague spatial concept, with using the *PlanGraph* to model various contextual factors related to the *SharedPlan* in communication. The GIS can reason about how to handle the context-dependency problem under constraints of the currently shared context represented in the *PlanGraph* during the human-GIS communication.

4.4.5 Design and Representation of Actions and Recipes

The system needs to have general knowledge of actions and their recipes involved in construction of *PlanGraph* before reasoning about how to handle the context-dependency problem with use of *PlanGraph*. We designed the GIS agent's general knowledge by following the human GIS operator's knowledge involved in human-human communication of vague spatial concepts, based on the human emulation principle.

Part of the human GIS operator's general knowledge is used in communication without involving the vagueness problem, and some of it is used in communication involving the vagueness problem. In the former case, the operator needs to have knowledge about: (1) how to understand the user's intention underlying the user information request; (2) how to collaborate with the user toward the user's spatial information request; (3) how to execute GIS commands to satisfy the user's request. In the later case, the GIS operator needs to know: (1) how to understand the meaning of a vague spatial concept with context information; (2) how to understand the context behind the user's spatial information request; (3) how to retrieve /infer context information from the current context; (4) how to repair the shared context and reduce vagueness through collaborative dialogues.

We also need to equip the GIS agent with general knowledge involved in these two cases by following the human GIS operator's knowledge described above. In human-GIS communication, we consider only two types of agents, the GIS agent and the user agent. In this domain, some actions can be performed by the GIS agent only, some of them can be carried out by the user agent only, and some of them can be done by either the GIS or the user. In design of actions and recipes for the GIS, we define the agent type responsible for the three classes of actions as G, U and $G_{i,}$ representing the GIS agent, the user agent, and either of them, respectively.

Actions and Recipes in Communication without Vagueness

Upon receiving the user's spatial information request to the GIS, the GIS agent needs to recognize the user's intention, *Spatial Intention*, underlying the request (see Figure **4-6** (a)). A typical example of *Spatial Intention* underlying the user's spatial information request is to see a map, *See Map* (see Figure **4-6** (b)).

(a) A general action representing the user's intention underlying the user's spatial information request to GIS

(b) An example action, See Map, representing the user's intention, to see a map, underlying the user's map request to GIS

Figure **4-6**: *Spatial Intention* Underlying User's Spatial Information Request

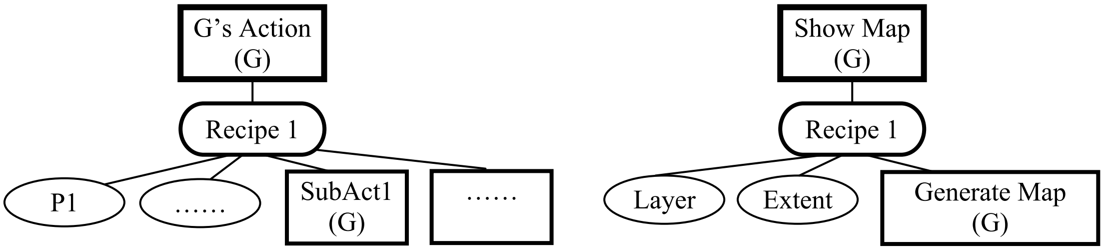
After understanding the user's *Spatial Intention* underlying the request, the GIS agent needs to know how to collaborate with the user to reach the user's intention as their common goal. Such knowledge can be designed as one of the recipes (Figure **4-7** (a)) for *Spatial Intention*. In human-GIS collaboration for *Spatial Intention*, the GIS agent and the user usually assume different responsibilities. The GIS agent usually needs to generate spatial information (*G's Action* in Figure **4-7** (a)) that satisfies the user's *Spatial Intention* and the user accepts it (*U's Action* in Figure **4-7** (a)). Therefore, *G's Action* is usually designed to be performed before *U's Action*. For example, one way for the GIS agent to help the user to reach the user's spatial intention, *See Map*, can be designed as the recipe in Figure **4-7** (b)). This figure shows that with this recipe, the GIS agent needs to show the map to the user (represented by the subaction *Show Map*) first, and then the user can see the map (represented by the subaction *Watch Map*).

(a) General recipe for GIS to help the user for the user’s intention as their common goal

(b) An example recipe for GIS to help the user for the user’s intention to see a map

Figure **4-7**: Recipes for GIS to Reach User’s Spatial Intention

To satisfy the user’s *Spatial Intention*, the GIS agent needs to know how to perform its own responsibilities, that is, *G’s Action*. A recipe for such an action usually involves various parameters, and certain subactions involving the execution of a GIS command (Figure **4-8** (a)). An example for *G’s Action* is shown in Figure **4-8** (b). This recipe for *Show Map* involves two parameters (*Layer* and *Extent*), and a GIS command related sub action *Generate Map*. In addition, the GIS agent also needs to have knowledge about how to instantiate the parameters involved in *G’s Action*, e. g. the two parameters in the recipe in Figure **4-8** (b). For more examples of *G’s Action*, see the actions and recipes in Appendix A.



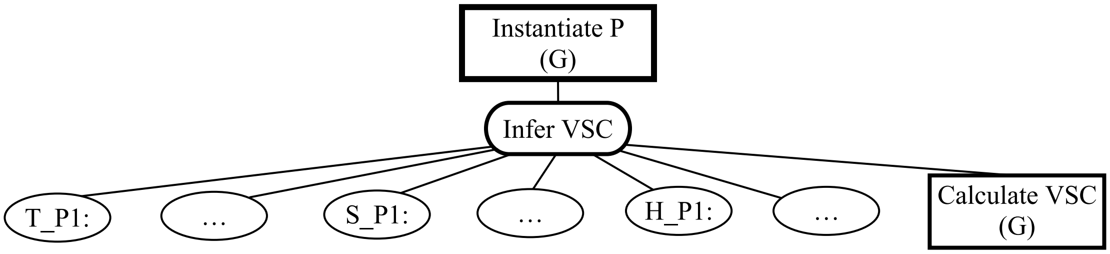
(a) General recipe for GIS to execute its action, *G's Action*.
One of SubActs is to execute a GIS command.

(b) An example recipe for the GIS agent's duty, *Show Map*
Layer: the map content
Extent: the map extent

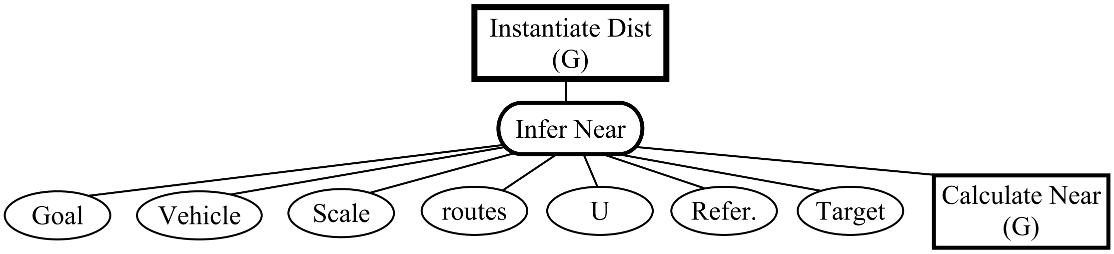
Figure **4-8**: Recipes for G's *Action*

Actions and Recipes in Communication Involving Vagueness

In this study, a vague spatial concept is usually used by the GIS to instantiate a parameter involved in a GIS command related action. For example, *near* in the user request of "I'd like to see the map of grocery stores *near Apartment A*" is used by the GIS to instantiate the parameter *Dist*, representing the spatial distance involved in certain GIS commands, e. g. a spatial query with use of the spatial distance. By following HCP1(c) and the human's context-dependent model about the meaning of a vague spatial concept *VSC* (Eq. **3.1**), we can design a general recipe, *Infer VSC*, for the GIS to instantiate a parameter from a vague spatial concept (Figure **4-9** (a)). In the recipe in Figure **4-9** (a), the parameters represent major contextual factors that influence the meaning of the vague spatial concept, and are designed as optional parameters. The subaction *Calculate VSC* in Figure **4-9** (a) gives the parameter value based on the system's model of the meaning of the vague spatial concept. The parameter values representing contextual information can be retrieved from or inferred based on the shared context represented in *PlanGraph*. With this recipe, the GIS can infer the meaning of a vague spatial concept by using context information based on the GIS agent's model about the vague spatial concept. An example recipe of *Infer VSC*, *Infer Near* for the GIS to understand the vague spatial concept *near* is shown in Figure **4-9** (b). The system can use this recipe to understand *near* based on its context-dependent model of the meaning of *near*.



(a) A general recipe for the GIS to instantiate a parameter P with a vague spatial concept VSC with major context information. *Calculate VSC* gives the parameter value based on the system's model of the meaning of the vague spatial concept.
T_Pi: parameter representing a task context factor; *S_Pi*: parameter representing a spatial context factor; *H_Pi*: parameter representing a human context factor

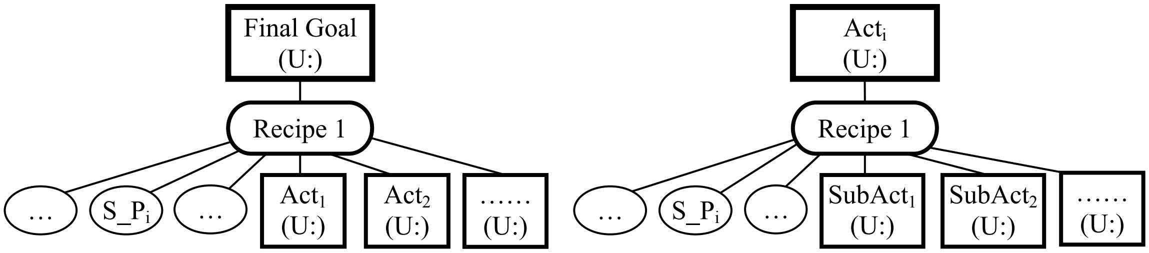


(b) A recipe for the GIS to calculate a distance value from the vague spatial concept *near* with major context information
Goal: user's goal behind the spatial information request;
Vehicle: user's transportation mode;
Scale: spatial size of the interested area;
U: user name
Refer.: Reference location;
Target: Target location
Routes: Spatial distribution of routes

Figure **4-9**: Recipe of *Infer VSC*

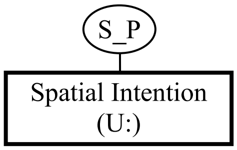
The GIS agent also needs to have knowledge about how to understand the context behind the user's spatial information request to the GIS by following HCP1(a). The general recipes in Figure **4-10** (a) and (b) show that the user's final goal (*Final Goal*) in the context behind the user's spatial information request can involve the user's activities

(*Act_i*) toward the goal and each user activity may involve sub activities (*$SubAct_i$*). The user's *Final Goal* and each activity (*Act_i*) or sub action (*$SubAct_i$*) in the activity may involve parameters (*S_P_i, S_P_j*) representing spatial knowledge pre-conditions, which are spatial context factors for the GIS agent to understand the meaning of a vague spatial concept. Thus, the user's *Spatial Intention* underlying the spatial information request to the GIS is usually designed for the user to identify the spatial information (modeled in parameters *S_P*) involved in the user's goal and the activities (Figure **4-10** (c)). For examples of actions and recipes related to different contexts, refer to Appendix A.



(a) A general recipe for the user's *Final Goal*. S_P_i Parameter representing a spatial context factor

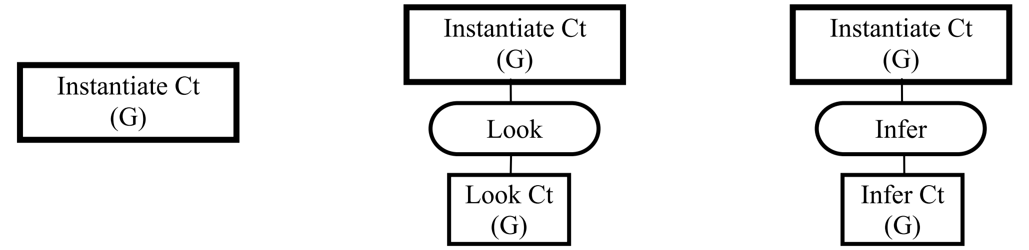
(b) A general recipe for *Act_i* toward *Final Goal*. S_P_j: Parameter representing a spatial context factor



(c) Relation between Spatial Intention and S_P in actions and recipes toward the user's Final Goal. S_Pi Parameter representing a spatial context factor.

Figure **4-10**: General Recipes Related to Context

The human communicator needs to retrieve such context information from or infer it based on the current shared context in human-human communication to understand the vague spatial concept with major context information. Similarly, the GIS agent also needs to have knowledge about how to retrieve or infer parameter values of the parameters in Figure **4-9** from *PlanGraph*. Therefore, the GIS agent needs to have knowledge specifically about how to instantiate these parameters. In this study, an action used to instantiate a parameter representing a context factor (*Ct*) is designed as *Instantiate Ct* (see Figure **4-11** (a)). Its recipe *Look* and *Infer* are designed by following the way by which the human user retrieves and estimates context information, respectively (Figure **4-11** (b, c)).



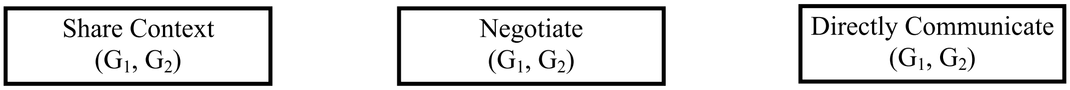
(a) Action for the GIS to instantiate a parameter *Ct* representing a context factor

(b) Recipe for the GIS to look for the parameter value for *Ct* from the existing *PlanGraph*

(c) Recipe for the GIS to infer the parameter value for *Ct*

Figure **4-11**: Actions and Recipes for Retrieving/Inferring Context Information

We can design a set of actions and their recipes for the GIS agent to handle the vagueness problem through various collaborative human-GIS dialogues, by following HCP2(c) and HCP2(d). For example, in Figure **4-12**, by following HCP2(c), *Share Context* is designed as a general action for the GIS agent to reach a shared context through context-sharing human-GIS dialogues; by following HCP2(d), *Negotiate* is designed as a general action for the GIS agent to refine the existing meaning of a vague spatial concept through negotiation human-GIS dialogues, and *Direct Communicate* as a general action for the GIS agent to further reduce vagueness through direct communication dialogues.



(a) Communicative action for the GIS and the user to share context information with each other

(b) Communicative action for the GIS and the user to negotiate the meaning of a vague spatial concept

(c) Communicative action for the GIS and the user to communicate the meaning of the vague spatial concept directly

Figure **4-12**: Communicative Actions to Handle the Context-Dependency Problem

4.4.6 Design and Representation of Mental States

The GIS agent's mental states on action/parameter nodes in the *PlanGraph* are designed by following the plan evolution process described in the *SharedPlan* theory (Grosz & Kraus, 1996, 1999), and further extended by following the HCPs, in particular, the GIS operator's mental states involved in handling the vagueness problem.

According to the plan evolution process described in the *SharedPlan* theory (Grosz & Kraus, 1996, 1999), a plan towards the goal/action α in collaboration usually evolves from partial to complete through several major steps (Figure **4-13**). We can summarize the agent's mental states (intention, commitment, and mental belief) required at each step of a plan to execute an action α in Table **4-1**. In this table, the "individual intention" means that the agent being modeled, that is, the GIS agent in this study, intends to achieve the success of α, and the "group intention" means that both the agent being modeled and the other agents in collaboration (e. g. the user agent in this study) all intend the success of α.

- Step 1: α is initially proposed as part of the collaboration;
- Step 2: α is agreed, intended and committed by all agents in collaboration;
- Step 3: The agent(s) responsible for α needs to start the plan completion process, which includes:
    - Step 3.1: If α can be directly executed, then the agent(s) assigned for α directly executes α,
    - Step 3.2: If α is complex and can not be directly executed,
        - Step 3.2.1: An appropriate recipe R for α is selected/constructed and an appropriate agent(s) is assigned for each parameter P_i and each sub action/goal β_i involved in this recipe if any;
        - Step 3.2.2: The agent(s) assigned for instantiation of each parameter P_i involved in R starts to instantiate P_i.
        - Step 3.2.3: The agent(s) assigned for each sub action/goal β_i involved in R if any begins to advance and evolve β_i toward the success of β_i.
- Step 4: The agent(s) responsible for α needs to monitor the result of execution of α, success or failure and updates its own belief and mutual beliefs about the result.

Figure **4-13**: Major Steps in Plan Evolution (Grosz & Kraus, 1999)

Table **4-1**: GIS Agent's Mental States on an Action α in *PlanGraph*

Step No.	Intention on α	Commitment on success of α	Belief (the agent being modeled believes that)
1	individual	individual	α is part of collaboration.
2	Individual & group	Individual and mutual	α is part of collaboration. All agents in collaboration also believe that α is part of collaboration.
3.1	Individual & group	Individual & mutual	α is basic, and needs to be executed directly; All agents in collaboration believe capabilities of the agent assigned for α to execute α.
3.2	Individual & group	Individual & mutual	α is complex, and needs to be elaborated and executed;
3.2.1	Individual & group	Individual & mutual	A recipe R for α is selected. All agents in collaboration agree with R selected for α.
3.2.2	Individual & group	Individual & mutual	Instantiation of each parameter P_i involved in R starts. All agents in collaboration believe capabilities of the agent assigned for instantiation of P_i
3.2.3	Individual & group	Individual & mutual	Further planning (& execution) for each sub-action/sub-goal β_i involved in R starts. All agents in collaboration believe capabilities of the agent assigned for further planning (& execution) of β_i
4	Individual & group	Individual & mutual	α is completed with the result of failure or success. All agents in collaboration need to know the result.

Note: For Step No. in this table, refer to Figure **4-13**

In the case that the plan for α is to instantiate a parameter *P* in *PlanGraph*, we can design the GIS agent's mental states on *P* by referring to the mental states required at each step of the plan α evolution (Table **4-2**).

Table **4-2**: GIS Agent's Mental States on a Parameter *P* in *PlanGraph*

Step No.	Intention on instantiation of p	Commitment on success of instantiation of p	Belief (the agent responsible for instantiation of p believes that)
1	individual	individual	P is waiting for instantiation.
2	Individual & group	Individual & mutual	P is waiting for instantiation. All agents in collaboration also believe that instantiation of p is necessary in collaboration.
3	Individual & group	Individual & mutual	P is being instantiated with a plan γ; All agents in collaboration believe capabilities of the agent(s) assigned for instantiation of p.
4	Individual & group	Individual & mutual	After Step 4, p is instantiated with the result of failure or success. All agents in collaboration need to know the result.

Note: For Step No. in this table, refer to Figure **4-13**

The human agent usually is not sure about the result of execution of an action α or instantiation of a parameter *P* involving vagueness until the other communicator(s) confirms the result. Similarly, the GIS agent is not sure if the result of α or *P* is success or failure until it receives the user's feedback to the result. At different stages of handling the vagueness problem or in different situations where the GIS agent and the user have different understandings about the context, the GIS agent can hold different mental beliefs on such results. For example, by following the HCP2(a), the GIS agent can hold different mental beliefs on the result of execution of an action α or instantiation of a parameter *p* involving vagueness before and after the shared context between the system and the user has been built. By further extending the agent's mental states in Step 4, we can design the GIS agent's mental states specifically on α or *P* involving vagueness. Table **4-3** and Table **4-4** show extended mental states on α or *P* involving vagueness in human-GIS communication, respectively. In Table **4-3**, the mental states required at Step 4.2, and Step 4.3 respectively represent the GIS agent's mental states on α before and after the shared context is built. This action is directly used by the GIS to understand the meaning of a vague spatial concept with context information. Similarly, in Table **4-4**, the mental states required at Step 4.2, and Step 4.3 respectively represent the GIS agent's mental states on *P* before and after the shared context is built. *P* is directly instantiated from a vague spatial concept. In this table, the mental state required at Step 4.2.1 is specifically designed to model the GIS agent's mental state on the result of a parameter *P* representing a contextual factor which is inferred with success and whose value may not be true. This mental state can also be used in a general case in which an optional parameter is inferred with success.

Table **4-3**: Extended Mental States on a Goal α Involving Vagueness

Step No.		Intention on α	Commitment on success of α	Belief (the GIS agent believes that)
4	4.1	individual, group	Individual & mutual	α is completed with the result of failure. All agents in collaboration need to know the result.
	4.2	Individual & group	Individual & mutual	α is completed with the result of success. All agents in collaboration need to know the result.
	4.3	Individual & group	Individual & mutual	The result of execution of the current plan for α may not be success because the context information used in this plan is not shared between the system and the user. All agents in collaboration need to know the result.
	4.4	Individual & group	Individual & mutual	The result of execution of the current plan for α may not be success although the context information used in this plan has been shared between the system and the user . All agents in collaboration need to know the result.
	4.5	Individual & group	Individual & mutual	The result of execution of the current plan for α may not be success because one of its descendants involves a parameter instantiated from a vague spatial concept. All agents in collaboration need to know the result.

Table **4-4**: Extended Mental States on a Parameter *P* Involving Vagueness

Step No.		Intention on instantiation of p	Commitment on success of instantiation of p	Belief (the agent responsible for instantiation of p believes that)
4	4.1	Individual & group	Individual & mutual	P is instantiated with the result of failure. All agents in collaboration know the result.
	4.2.1	Individual & group	Individual & mutual	P's value is inferred with the result of success. All agents in collaboration know the result.
	4.2.2	Individual & group	Individual & mutual	P is instantiated with the result of success. All agents in collaboration know the result.
	4.3	Individual & group	Individual & mutual	The result of instantiation of p may not be success because the context information used in the current plan for instantiation of p is not shared between the system and the user. All agents in collaboration know the result.
	4.4	Individual & group	Individual & mutual	The result of instantiation of p may not be success although the context information used in the current plan for instantiation of p has been shared between the system and the user. All agents in collaboration know the result.
	4.5	Individual & group	Individual & mutual	The result of instantiation of p may not be success because one of its descendants involves a parameter instantiated from a vague spatial concept.

4.4.7 Design and Representation of Reasoning Algorithms

Upon receiving each user input, the GIS agent needs to react to the input during the collaborative human-GIS planning process with three steps, including interpretation, elaboration and response control (Balkanski & Hurault-Plantet, 2000; Lochbaum, 1998) (Figure **4-14**). At the step of interpretation, the system needs to interpret the meaning of the new input. At the step of elaboration, the system needs to advance the existing human-GIS collaborative plan toward its success, and execute whatever can be executed toward its success. At the step of response control, the system needs to control when and how to return the responses (maps/questions/answers) that have been generated so far to the user interface.

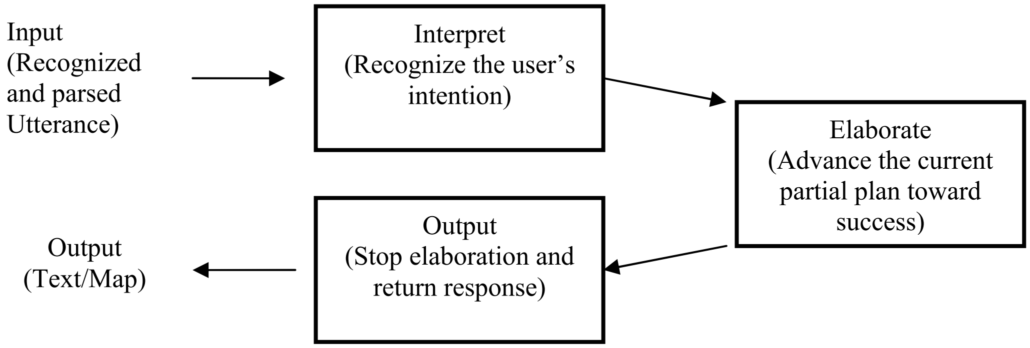


Figure **4-14**: Three Major Steps for the GIS to React to User's Utterances

PlanGraph can facilitate the GIS agent's reasoning involved in the three steps to communicate with the user. This section describes the reasoning algorithms associated with *PlanGraph*, which are designed for the GIS agent to manage human-GIS communication in these three major steps.

Interpretation Algorithms

The Interpretation Algorithms (**IA**) in the *PlanGraph* model are designed based on those in the *RGraph* model (Lochbaum, 1994, 1998; Lochbaum et al., 2000). By following the interpretation reasoning algorithms developed in *Rgraph*, in this study we also model interpretation of the user's input utterance as a process by which the GIS agent decides how the user's intention underlying the input utterance contributes to the existing *PlanGraph*, which represents the GIS agent's knowledge of the existing *SharedPlan* between the GIS and the user.

In human-GIS communication, we assume that the user's input utterance usually is short and expresses only one major intention α. Under this assumption, **IA**s in the *PlanGraph* model are designed for the following two cases (Balkanski & Hurault-Plantet, 2000): 1) the input utterance is an initial request from the user and *PlanGraph* is empty, and 2) *PlanGraph* is not empty and the user input contributes to the current *SharedPlan* between the GIS and the user:

- In the first case, if the GIS agent believes that it can participate in α, the GIS agent needs to initiate the *PlanGraph* with α, update its mental state on α, and focuses its attention on α.
- In the second case, the GIS agent needs to find a way to integrate α with the existing *PlanGraph* and updates its mental states and attention focus in *PlanGraph*. α can be directly contributing to the attention focus β in *PlanGraph*, e. g. α is β or part of β; β is part of α; α represents the user's mental attitude on β. α can also be related to other components in *PlanGraph*.

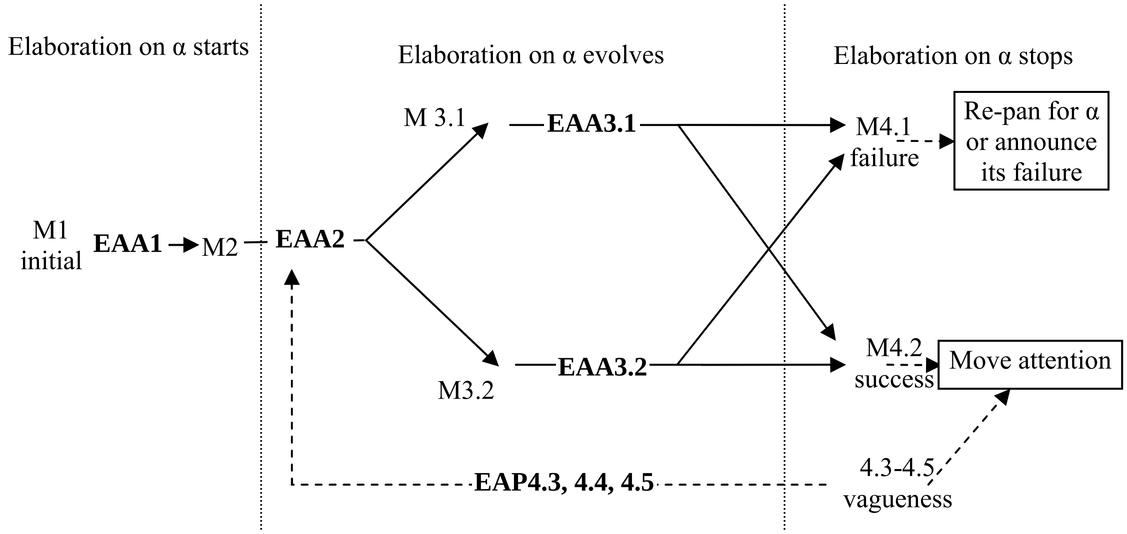
For examples of **IAs**, see Appendix C. These algorithms can be further extended for the GIS agent to handle vagueness by following how the human communicator interprets the other communicator's input in human-human communication involving vagueness.

Elaboration Algorithms

The Elaboration Algorithms (**EA**s) in the *PlanGraph* model are designed by following the plan evolution process (Figure **4-13**) and referring to the task-advancement reasoning algorithms in Balkanski and Hurault-Plantet's work (Balkanski & Hurault-

Plantet, 2000). At each step of the plan evolution process (Figure **4-13**), if the agent(s) responsible for that plan holds mental states required at that step, the agent can evolve the plan (e. g. change the mental state on that plan and/or execute action(s) involved in that plan) to the next step towards the success of the plan. The *PlanGraph* records the GIS agent's mental states on each action/parameter in the *SharedPlan* built by the GIS and the user and the attention focus on the *SharedPlan*. Therefore, by following the plan evolution process described above, we can design **EAs** for the GIS agent to reason about how to evolve the plan for executing the focus action or instantiating the focus parameter with use of the *PlanGraph*. The plan evolution is a recursive process which interleaves planning (e. g. finding a suitable recipe for a complex action involved in the collaboration) and execution (e. g. execution of basic actions and instantiation of parameters). Thus, the **EAs** are designed also as such a recursive process.

The elaboration process is designed as always starting with elaboration of the focus action/parameter in the *PlanGraph*, e. g. an action to be done or a parameter to be instantiated. In case the GIS agent's current attention focus in *PlanGraph* is on an action α, the Elaboration Algorithms on an Action α (**EAA**s) are designed for the GIS agent (G) to evolve α from partial to complete, step by step, as follows (Figure **4-15**):



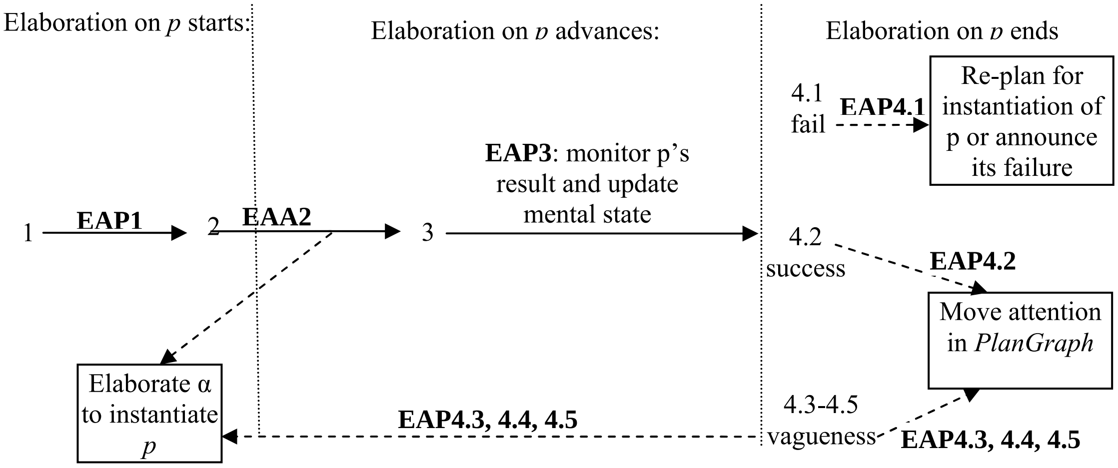
M1, M2 … The GIS agent's mental state at each step during the α evolution process **EAA1**, **EAA2**, …. **EAA** at each step

Figure **4-15**: Structure of **EAAs**

- **EAA** at Step 1: if G holds mental states required at Step 1, G can start to evolve α toward Step 2 by evolving its mental state on α to the mental state required at Step 2
- **EAA** at Step 2: if G holds mental states required at Step 2, then the agent(s) can start to evolve α toward Step 3 by evolving its mental state on α to the mental state required at Step 3.1 or Step 3.2.1 after determining that α is basic or complex (and if α is complex, an appropriate recipe needs to be selected for α);
- **EAA** at Step 3:
  - **EAA** at Step 3.1: If G holds the mental state required at Step 3.1 (α is basic), G directly executes α, monitors the result of the execution of α and then updates its mental state required at Step 4;

  - **EAA** at Step 3.2
    - **EAA** at Step 3.2.1: If G holds mental states required at Step 3.2.1 (α is complex and a recipe R is selected for α), the agent G_i responsible for each parameter P_i in R starts to elaborate P_i if any and updates its mental state on α as required at Step 3.2.2,
    - **EAA** at Step 3.2.2: If G holds mental states required at Step 3.2.2, G has all parameters in the current plan R for α instantiated, the agent G_i responsible for each sub action β_i in R starts to elaborate β_i and G updates its mental state required at Step 3.3.3.
    - **EAA** at Step 3.2.3: If G holds mental states required at Step 3.2.2, G has all sub actions in the current plan R for α finished, monitors the result of execution of R and updates its mental state on α required at Step 4.;
- **EAA** at Step 4: If G holds a mental state required at Step 4, G needs to give up the current plan for α if α fails, or needs to move attention to the next action in the collaboration that G can participate in if α succeeds.

Similar to design of the **EAA**s, the Elaboration Algorithms on Parameters (**EAPs**) are designed by referring to the evolution process of the plan α for the agent(s) G to instantiate the focus parameter *p* as follows (Figure **4-16**):



1, 2, 3……Mental States required at each step **EAP1**, **EAP2**, …. EAP at each step

Figure **4-16**: Structure of **EAPs**

- **EAP** at Step 1: if G holds mental states required at Step 1, then G can start to evolve *p* toward Step 2 by evolving its mental state on *p* to the mental state required at Step 2.
- **EAP** at Step 2: if G holds mental states required at Step 2, G can start to evolve *p* toward Step 3 by proposing a plan α to instantiate *p*;
- **EAP** at Step 3: if G holds mental states required at Step 3, G waits for α's execution result, updates the parameter value for *p* and the mental state required at Step 4 depending on the result from α.
- **EAP** at Step 4: If G holds a mental state required at Step 4, G needs to give up *p*'s instantiation if α fails, or needs to move its attention to the next action/parameter in the collaboration that G can contribute to if α succeeds.

From the **EAA**s and **EAP**s described above, we can see that by applying the **EAA**s or **EAP**s, the GIS agent can evolve its mental states on an action or a parameter from partial to complete. Each **EAA/EAP** for the focus action α/parameter *p* also describes conditions (mainly the GIS agent's mental states on α/*p*) under which it applies, and the effects (mainly the GIS agent's mental state change on α/*p*, attention change in *PlanGraph*, and execution of α or instantiation of *p*) that it will bring to α/*p*.

By following the HCPs involved in handling the vagueness problem, we can extend **EAAs**/**EAPs** for the GIS to elaborate the focus action α involving vagueness or the focus parameter *p* instantiated from a vague spatial concept as the GIS agent's mental states are extended for handling the vagueness problem. Following are examples of **EAAs and EAPs** at Step 4.3, Step 4.4, and Step 4.5 by referring to the mental states designed in Table **4-3** and Table **4-4**, respectively:

- **EAA/EAP** at Step 4.2, Step 4.3, and Step 4.5 in case α/instantiation of *p* is just finished: If G holds a mental state required at Step 4.2, Step 4.3, or Step 4.5, G needs to move its attention to the next action/parameter in the collaboration that G can contribute to. In human-human communication, the human communicator usually continues the collaboration with an inferred meaning of a vague spatial concept. By utilizing following the human's solution in this case, we design extended **EAAs** and **EAPs** here.
- **EAA/EAP** at Step 4.2, Step 4.3, or Step 4.5 in case the result of α/ instantiation of *p* is disputed by the user: G holds a mental state required at Step 4.2, Step 4.3, or Step 4.5, G needs to re-visit α or the plan to instantiate *p*. In human-human communication, the human communicator usually tries to improve the previous plan involving vagueness when the user disagrees with

the result of the previous plan. By adopting this human solution, we have designed **EAAs** and **EAPs** presented here. These algorithms can be further detailed by employing the human solutions in this case. For example, by using HCP2(c) and HCP2(d), we can design **EAAs**/**EAPs** in this case for the GIS agent to repair the shared context and further reduce vagueness through use of various collaborative dialogue strategies.

Here is another example for extension of **EAAs**/**EAPs** for the GIS to handle the context-dependency problem by utilizing HCP1(b):

- **EAA** at Step 3.2.1 in case the current plan for α is used to understand a vague spatial concept with contextual information: If G holds mental states required at Step 3.2.1 (α is complex and a recipe R is selected for α), the agent G_i responsible for each parameter P_i representing a contextual factor in R needs to infer the context behind the user's request involving the vague spatial concept before starting to elaborate P_i if any and updates its mental state on α as required at Step 3.2.2.

Response Control Algorithms

The output reasoning algorithms in the *PlanGraph* model are designed by following the plan evolution process and referring to the stopping conditions in the task-advancement reasoning algorithms in Balkanski and Hurault-Plantet's work (Balkanski & Hurault-Plantet, 2000). During the plan evolution process, the agent(s) responsible for that plan may need to communicate with other collaborators in different situations, e. g. exchanging mental beliefs/commitments on that plan or executing a communicative action (e. g. asking questions or answering questions) in that plan. So, with the *PlanGraph* modeling the *SharedPlan* between the GIS and the user, the GIS agent can know whether the plan/subplan involved in the *SharedPlan* is communicative or not, or whether there is a need to communicate with the user or not. Therefore, we can design output reasoning algorithms for the GIS to reason about when to stop elaboration and return responses to the user by using the *PlanGraph*.

In human-GIS communication, we currently consider two cases for the GIS agent to stop elaboration and output responses: 1) in case execution of an action in the *PlanGraph* involves communication, e. g. in case the GIS needs to execute a basic action α representing the GIS agent's question to the user; 2) in case the GIS agent needs to communicate with the user about certain mental states, e. g. mutual beliefs, mutual commitments. Two Output Response (**OR**) algorithms can be designed respectively for these two cases. In these two cases, the GIS agent needs to send out responses, updates its mental states and attention focus in *PlanGraph*. For examples of **OR**s, refer to Appendix C.

4.4.8 Comparison with the *Rgraph* model

The *PlanGraph* model extends the *Rgraph* model based on the computational collaborative discourse theory and the *SharedPlan* theory, and is further extended by following the HCF, in particular, HCPs involved in handling the vagueness problem. Here extensions of the *PlanGraph* model with respect to the *Rgraph* model are summarized as follows.

Firstly, *PlanGraph* models not only action nodes, but also parameter nodes. In *Rgraph*, parameters are modeled as knowledge pre-conditions that must be satisfied before execution of plans involving the parameters. However, parameters are not explicitly represented in the *Rgraph*. The vague spatial concept is the major focus in this study. The GIS agent's understanding about it usually involves use of complex plans. Thus, the *PlanGraph* is designed to record parameter nodes explicitly for convenience of reasoning about how to instantiate such parameters with complex plans.

Secondly, *PlanGraph* models not only mental statuses on actions, but also those on parameters. In addition, *PlanGraph* also extends *Rgraph* by incorporating a set of mental states involved in handling the vagueness problem. *Rgraph* does not explicitly model the agent's mental states on parameter values. In this study, the GIS agent may hold more mental states on the result of an action or the instantiation result of a parameter involving vagueness than just simple success or failure. Therefore, for convenience of reasoning about how to handle the vagueness problem, *PlanGraph* is designed to record the GIS agent's mental states on both action and parameter nodes, and incorporate these mental states involved in handling the vagueness problem.

Thirdly, by following the plan evolution process (Grosz & Kraus, 1996, 1999), the *PlanGraph* model also extends the *Rgraph* model by providing reasoning algorithms for not only interpreting each utterance in discourse, but also elaborating the *SharedPlan* underlying the collaborative discourse and controlling responses in the discourse. In particular, the **EAAs** and **EAPs** in the *PlanGraph* model enable both the planning process and the actual execution process. These reasoning algorithms in the *PlanGraph* model are further extended for the GIS agent to handle the vagueness problem by using the HCF.

4.5 Example Human-GIS Collaborative Dialogue

This section explains a hypothetical collaborative dialogue between a spoken natural language GIS incorporating the collaborative dialogue approach and the *PlanGraph* model and a human user (Figure **4-17**). This dialogue example is used to illustrate how the collaborative dialogue approach and the *PlanGraph* model can help the GIS agent to handle the context-dependency problem. For convenience of description, let the user be a male. For all actions and recipes involved in this example, refer to Appendix A. In all *PlanGraphs* shown in this section, the plan evolution step number at the right side of an action/parameter node represents the GIS agent's mental states required at that step. For each plan evolution step number and mental states required at that step, refer to Table **4-1**, Table **4-2**, Table **4-3** and Table **4-4**.

G: GIS U: User-A

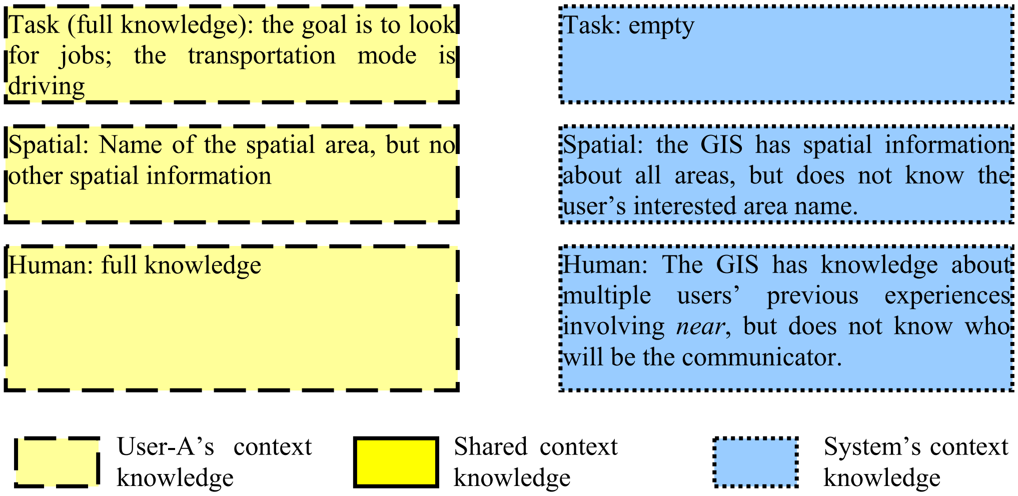
Before the following conversation, the map on the GIS is empty.

U (1): Hi, I am User-A. I would like to see cities near City A.
G (1): (show a map of City A, including interstates, and highlight certain cities near City A) For the plan of vacation purpose by driving, here are cities within 300 miles of City A. Is this OK?
U (2): No.
G (2): What is your goal in asking for this map?
U (3): I am planning to look for jobs in cities near City A. I plan to work there by driving from City A.
G (3): (Highlight fewer cities near City A) Here are cities within 150 miles of City A, is this OK?
U (4): Yes. Thanks.

Figure **4-17**: Hypothetical Collaborative Human-GIS Dialogue

4.5.1 Building Initial Shared Context through U (1) and G (1)

Before the dialogue shown in Figure **4-17** starts, the GIS map is empty. As to the shared context involved in communication (Figure **4-18**), the GIS and the user share only common sense knowledge about understanding the language and conventions involved in human-GIS communication/collaboration. The user holds full knowledge about the task context and the human context, but has no knowledge about the spatial context to be involved in the task domain. The system does not have knowledge about who will be the user and which spatial area and associated spatial information will be the user's interest. However, the system holds geospatial information needed for the conversation in its database, and also holds part of the user's previous experiences about understanding *near* in various contexts.



Task (full knowledge): the goal is to look for jobs; the transportation mode is driving

Task: empty

Spatial: Name of the spatial area, but no other spatial information

Spatial: the GIS has spatial information about all areas, but does not know the user's interested area name.

Human: full knowledge

Human: The GIS has knowledge about multiple users' previous experiences involving *near*, but does not know who will be the communicator.

Figure **4-18**: Shared Context before Communication

Interpretation of U (1)

After receiving the user's first utterance U (1): "Hi, I am User-A. I would like to see cities near City A.", the system initiates *PlanGraph* (Figure **4-19**) with the user's *Spatial Intention* under U (1) as *See Map* by applying the **IAs** in the first case and focuses on this action. This is because it believes that it can help the user to achieve this spatial intention.

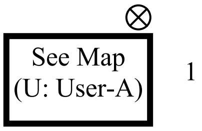


Figure **4-19**: *PlanGraph* Initiated after U (1)

Starting Elaboration after U (1)

Following interpretation of the user input in U (1), the GIS agent starts to elaborate the current *PlanGraph* towards the success of the user's spatial information request. The elaboration starts with the current attention focus in the *PlanGraph*.

By applying the recursive **EAAs** and **EAPs** in the *PlanGraph* model, the GIS elaborates the plan *See Map* in *PlanGraph* with an appropriate recipe for it (e. g. Figure **4-7** (b)), and further recursively elaborates all its descendants until the system needs to instantiate the parameter *Dist* required in the GIS command related sub plan, *Select by Distance* (Figure **A-8** (i) in Appendix A). By adopting the HCP1(a), the GIS agent uses a recipe *Infer Near* (Figure **4-9** (b)) to infer the meaning of the vague spatial concept *near* with context information first. The current *PlanGraph* is shown in Figure **4-20**.

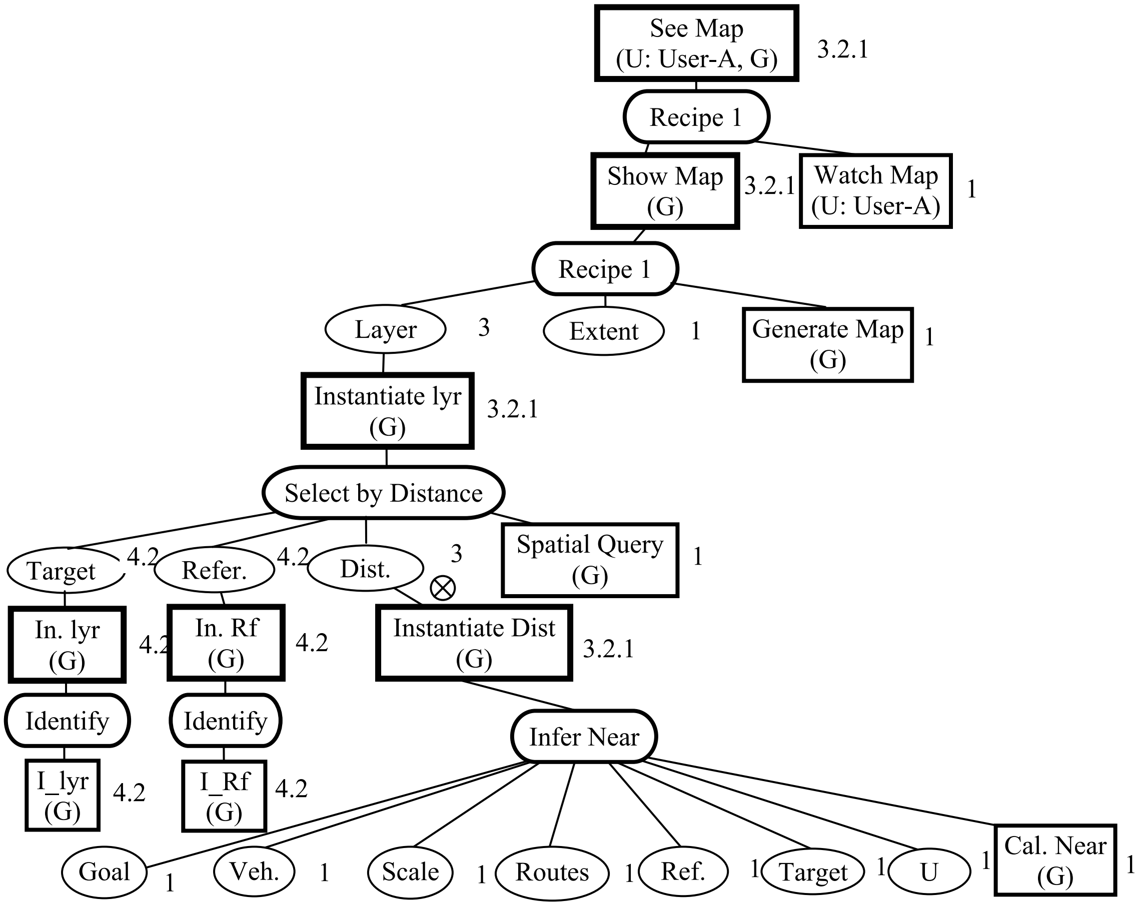


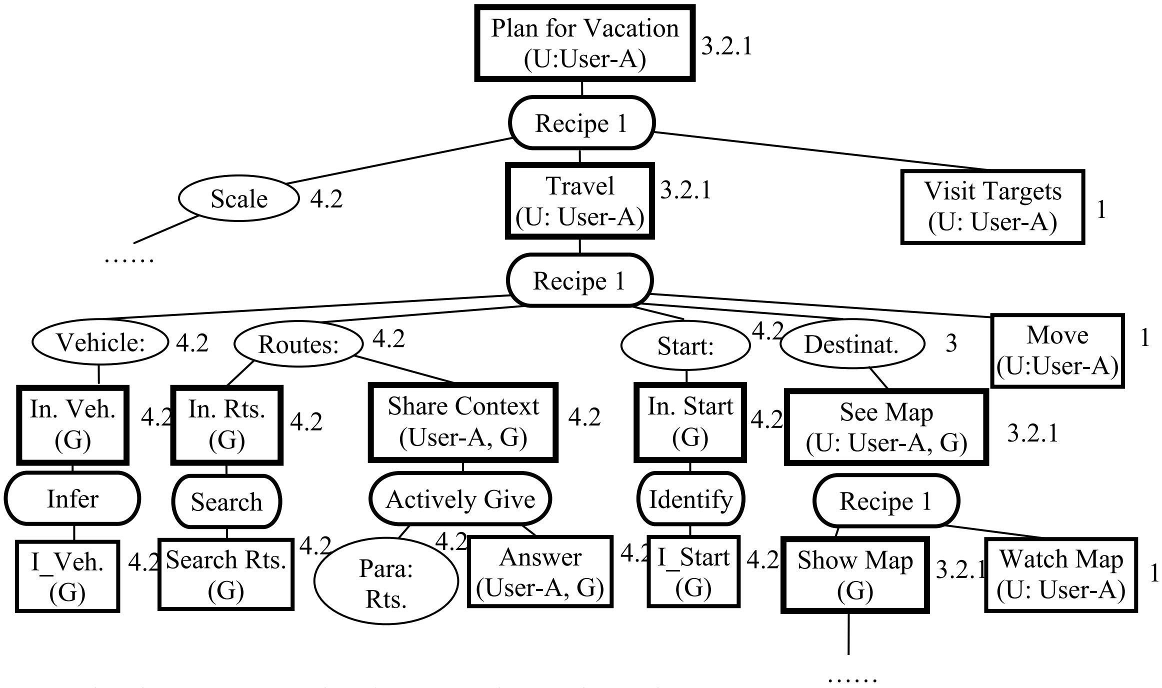
Figure **4-20**: *PlanGraph* before Executing *Infer Near*

Inferring Context before Understanding Near with Contextual Information

By applying **EAAs**/**EAPs** designed by following the HCP1(b), the GIS agent infers the context behind the user's intention in U (1) before executing the plan *Infer Near*. Based on its best knowledge, the system infers that the context behind U (1) is related to the goal *Plan for Vacation* (Figure **A-2** (a)). The system believes that the user's intention *See Map* is to identify spatial information about the traveling destinations for the goal *Plan for Vacation*. Then, the actions and recipes related to *Plan for Vacation* are integrated with the existing *PlanGraph* by linking *See Map* with *Destination*, a descendant parameter of *Plan for Vacation*.

Actively Sharing Spatial Contextual Information with the User

To facilitate building a shared context, the GIS agent prepares spatial contextual information needed in *Plan for Vacation*, e. g. routes needed in the traveling activity, with the plan *Actively Give* of *Share Context* (Figure **A-11** (c)) for the user and plans to actively share such spatial information with the user through the GIS map later. The current *PlanGraph* is shown in Figure **4-21**.



For missed components under *Show Map*, please refer to Figure **4-20**

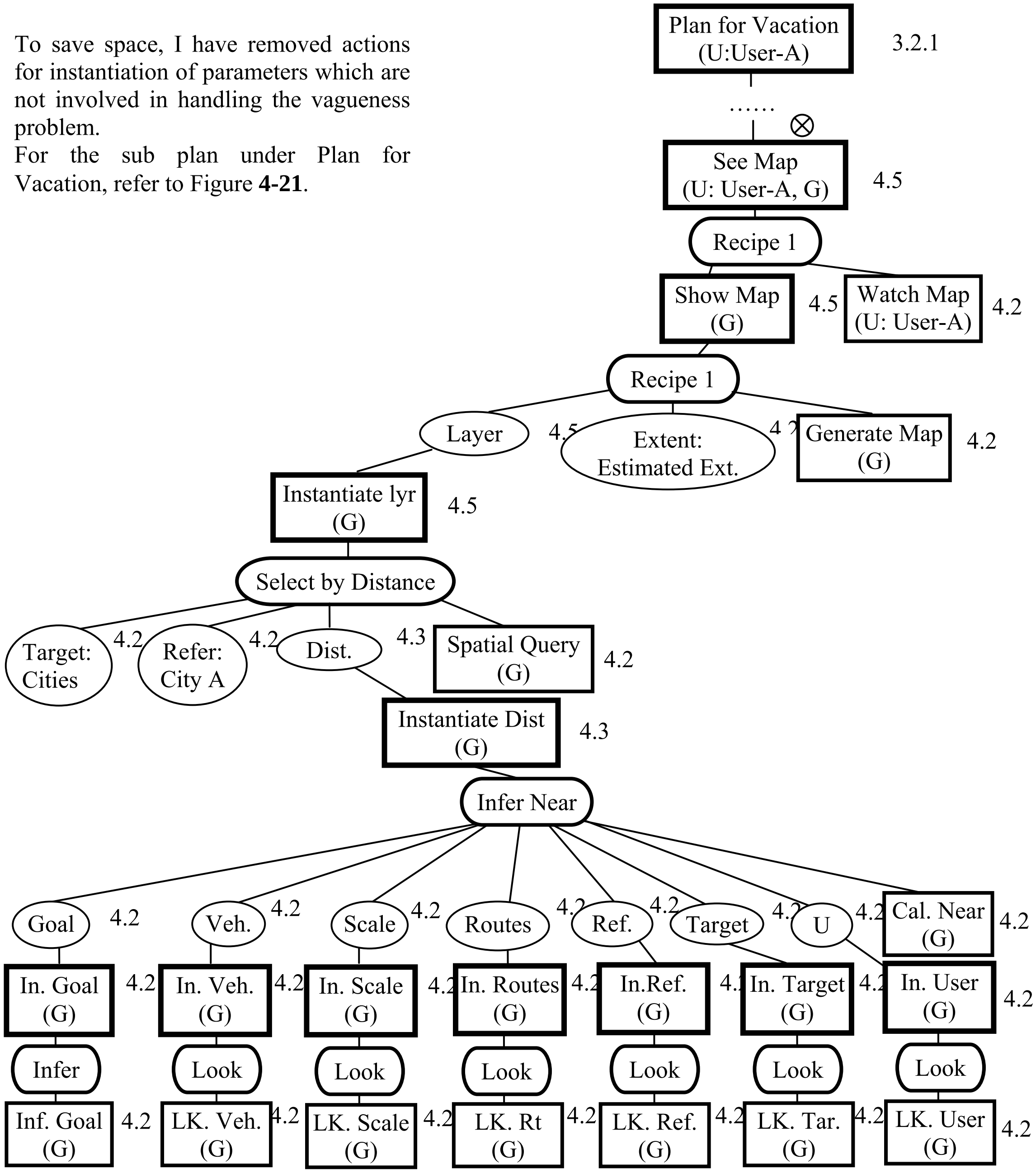
Figure **4-21**: *PlanGraph* after Inferring the Context

After inferring *Plan for Vacation*, the GIS agent comes back to continue to elaborate the plan *Infer Near*. By using the action and recipes shown in Figure **4-11**, the GIS agent infers/retrieves contextual information from the existing *PlanGraph* in Figure **4-21** that is needed in *Infer Near*, and executes this plan by assigning 300 miles to the parameter *Dist*. The system further generates a map response to the user in G (1) by finishing the plan *Show Map*. This map shows not only "cities within 300 miles of *City A*", but also the interstate distribution in the spatial area that the user is interested in. The interstate distribution information has been generated from the system's active context-sharing plan, *Actively Give*.

By applying the **EAAs/EAPs** designed for elaboration at Step 3 and 4, the GIS agent shifts its attention to *Watch Map* responsible by the user and updates its mental states on all action/parameter nodes in *PlanGraph* correspondingly. Because part of the context-information used in *Infer Near* is inferred, the GIS agent updates its mental states on *Instantiate Dist* and *Dist* as those required at the plan evolution Step 4.3. It also updates its mental states on the ascendants of *Dist* as those required at the plan evolution Step 4.5.

Controlling Responses to U (1)

The current attention focus *Watch Map* is to be done by the user. The GIS assumes that the user can successfully accomplish this action. Under this assumption, by applying the **ORs** designed, the system can shift its attention to the parent action of *Show Map*, *See Map*, which is the last action in which the system can participate in human-GIS collaboration after sending out responses in G (1). The current *PlanGraph* is shown in Figure **4-22**. The map response in G (1) is shown in Figure **4-23**.



To save space, I have removed actions for instantiation of parameters which are not involved in handling the vagueness problem.
For the sub plan under Plan for Vacation, refer to Figure **4-21**.

Figure **4-22**: *PlanGraph* after G (1) in the Hypothetical Dialogue Example

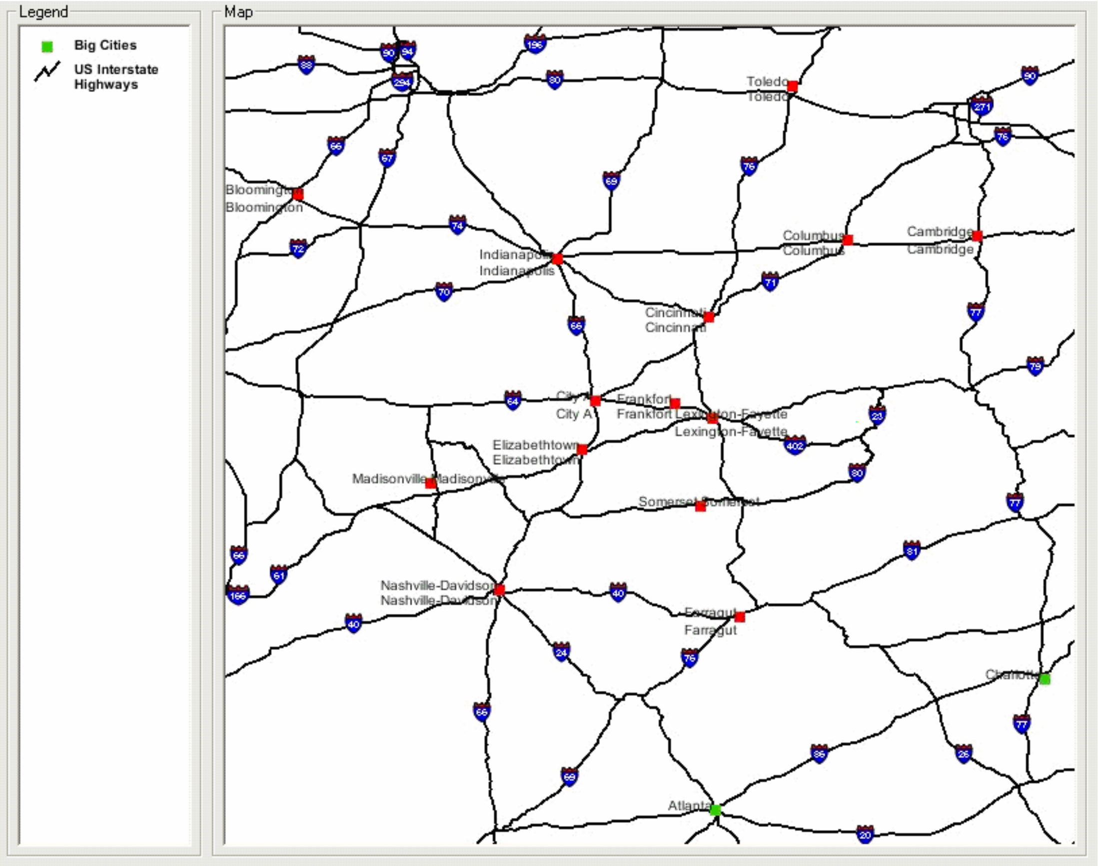


Figure **4-23**: Map Response in G (1) in Hypothetical Human-GIS Dialogue

After receiving the system's response in G (1), the user updates his context knowledge, in particular, on the spatial context. He agrees with the spatial area and road distribution information, but not the *near* distance. After U (1) and G (1), the shared context between the user and the system (Figure **4-24**) is improved with better sharing of spatial context and human context.

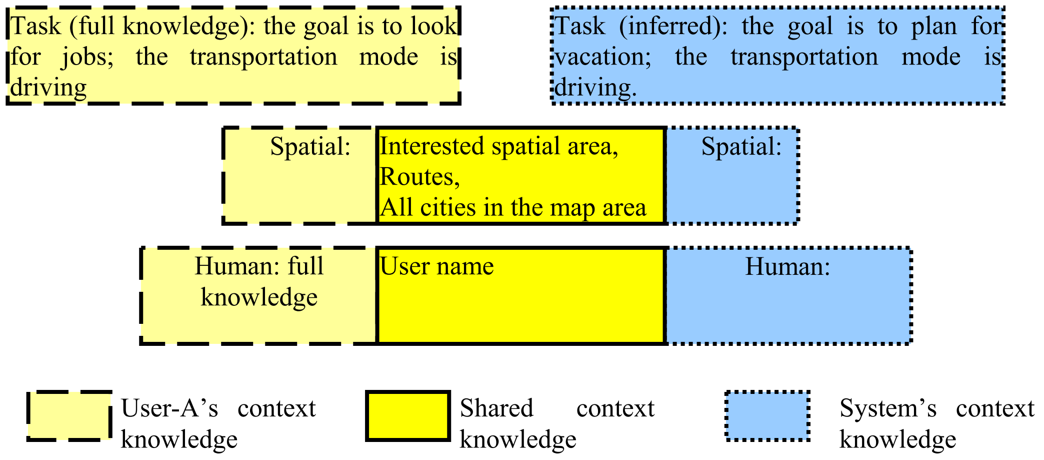
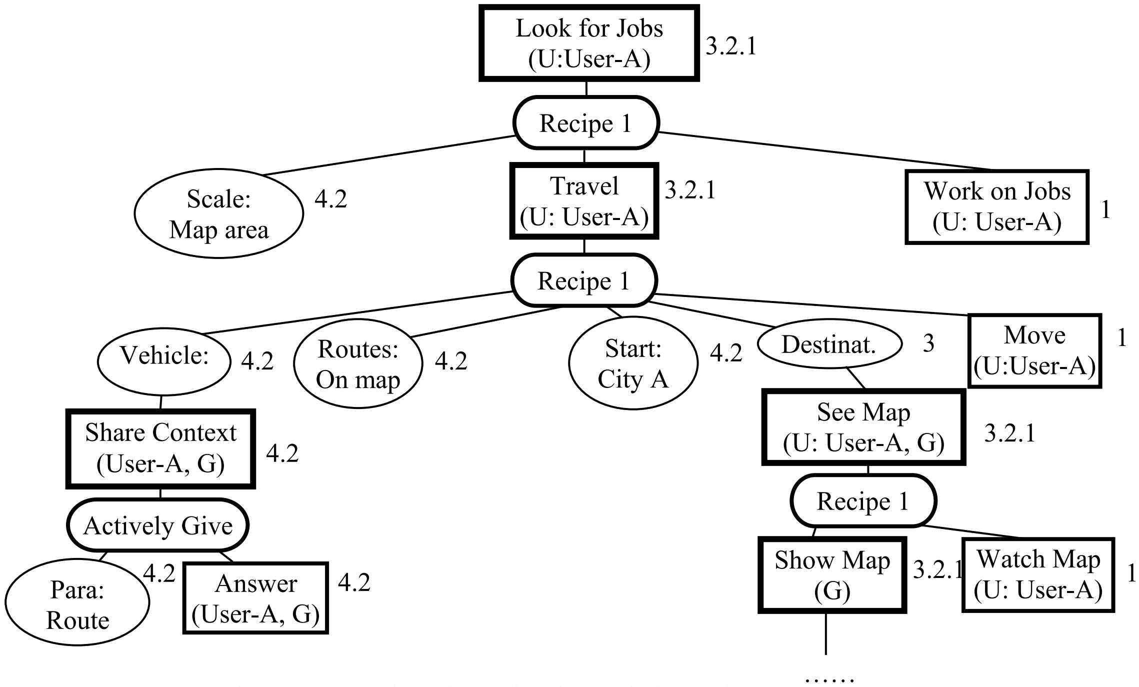


Figure **4-24**: Shared Context after Communication between U (1) and G (1)

4.5.2 Repairing Shared Context through G (2) and U (3)

After receiving the user's negative response in U (2), the GIS agent revisits the previous plan for *See Map* and all its descendants involving vagueness, in particular, the sub plan *Infer Near*, by applying **EAAs** and **EAPs** at Step 4.2, 4.3, and 4.5 (specifically designed for handling the vagueness problem). Because the user's final goal in the event behind U (1) is inferred in the previous plan *Infer Near*, the system uses the plan *Ask* of *Share Context* (Figure **A-11** (b)) to obtain such contextual information from the user in order to repair the shared context. So, the system asks the user in G (2): "What is your goal in asking for this map?". The user collaboratively answers this question in U (3): "I am planning to look for jobs in cities near City A.". In addition, the user also actively shares his transportation mode information with the system in U (3) without being asked: "I plan to work there by driving from City A". The system understands the later part of U (3) as *Actively Give* of *Share Context* (Figure **A-11** (c)).

With new updated contextual information from the user, the system re-elaborates the context behind U (1), and integrates the plan *Look for Jobs* (Figure **A-4**) with the existing *PlanGraph*. The system's updated context knowledge in the *PlanGraph* is shown in Figure **4-25**.



To save space, I have removed actions for instantiation of parameters that have the same values as those in Figure **4-22**

Figure **4-25**: System's Updated Context Knowledge after U (3)

After receiving U (3), the system and the user have reached a shared understanding about the current context (Figure **4-26**), which influences the meaning of *near* in both agents' model of *near*.

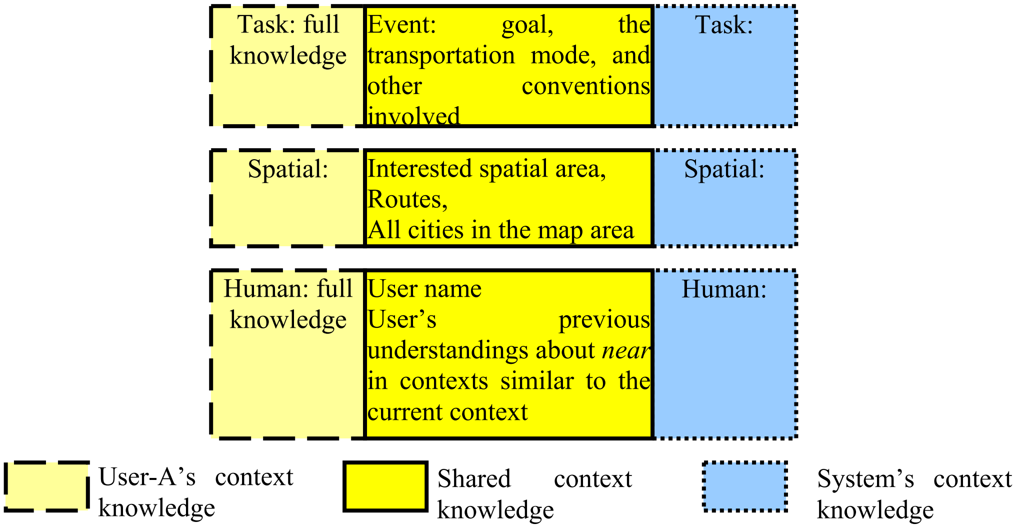


Figure **4-26**: Shared Context after Communication between U (1) and U (3)

4.5.3 Reaching Shared Meaning of *Near* under Constraints of Shared Context

By successfully building a shared context with the user through collaborative dialogues, the system re-infers the meaning of *near* with the plan *Infer Near*. This plan is improved with new contextual information retrieved from the plan *Look for Jobs* in the existing *PlanGraph* in Figure **4-25**. The system's new understanding about *near* is 150 miles of driving distance.

After re-inferring the meaning of *near*, the GIS agent re-generates a map response showing cities within 150 miles of driving distance to City A in G (3) (Figure **4-27**). Similarly, the GIS agent updates its mental states on *Dist* and *Instantiate Dist* as those required at the plan evolution Step 4.4 because it believes that all context information used for its understanding about the vague spatial concept *near* has been shared with the user.

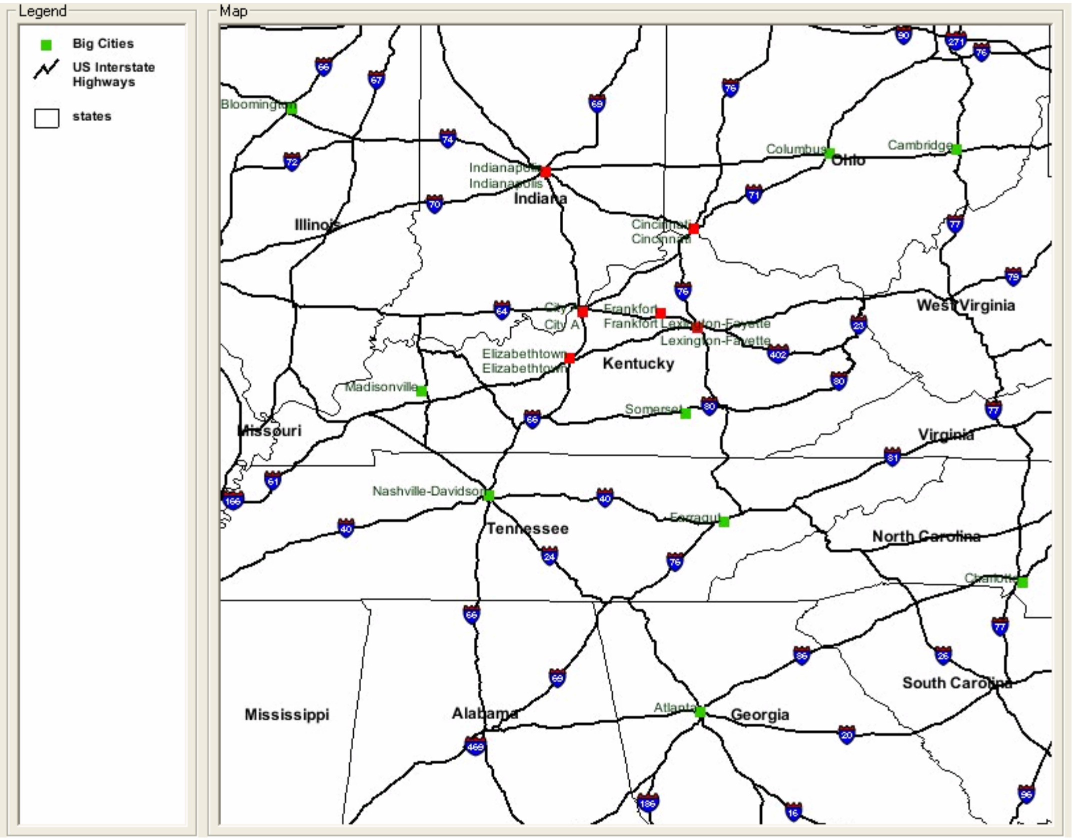


Figure **4-27**: Map Response in G (3) in Hypothetical Human-GIS Dialogue

Finally, the system's new understanding about *near* under the shared context built receives the user's agreement in U (4). Thus, they reach a shared understanding about the vague spatial concept *near* through the collaborative dialogue approach and the *PlanGraph* model.

4.6 Summary and Discussion

This chapter describes the collaborative dialogue approach and the agent-based computational model, the *PlanGraph* model in detail. This approach is developed for the

GIS to handle the context-dependency problem with collaborative human-GIS dialogues. This model is developed to support this approach.

The HCF enables us to understand the vagueness problem in human-GIS communication more comprehensively. It helps us to understand three major sources of vagueness in human-GIS communication. It also helps for understanding of three major dimensions of the shared context in human-GIS communication, that is, the three major types of contextual factors that influence the meaning of a vague spatial concept. Contextual knowledge related to the three types of contextual factors is usually distributed between the user and the system at the beginning of human-GIS communication. The distributed nature of the context knowledge in human-GIS communication leads to the need for building a shared context in human-GIS communication, which can further help the two agents to for reach closer understandings about the vague spatial concept. The need for a shared context leads to the need for collaborative human-GIS dialogues.

The need for collaborative human-GIS dialogues leads to our development of the collaborative dialogue approach. The major design principle of this approach is the human emulation principle, i. e. enabling collaborative human-GIS dialogues by emulating collaborative human-human dialogues. There are two keys for the GIS agent to handle the context-dependency problem with the collaborative dialogue approach. One is to enable the GIS to keep track of the shared context between the two agents; the other is to provide the GIS with reasoning capabilities involved in handling the context-dependency problem through collaborative human-GIS dialogues under the constraints of the shared context. The *PlanGraph* model is designed for implementation of the collaborative dialogue approach. This model includes two major components, the *PlanGraph* and associated reasoning algorithms, which support the two keys of the collaborative dialogue approach. The *PlanGraph* models the GIS agent's dynamic knowledge kept on the dynamic shared context. By modeling various contextual factors as its different components, *PlanGraph* can enable the GIS to keep track of the current state of the shared context, including the task context, the spatial context and the human context involved in the shared context. The associated reasoning algorithms are designed by following HCPs from the HCF and provide the GIS agent with reasoning capabilities needed.

The hypothetical human-GIS dialogue explained at the end of this chapter shows how the *PlanGraph* model can help the GIS agent to handle the context-dependency problem with the collaborative dialogue approach. With the context knowledge represented in the *PlanGraph* and reasoning algorithms designed for repairing the context, the GIS agent can reach a shared understanding about the current context with the user through various collaborative human-GIS dialogues. The shared context built between the system and the user helps the two agents to reach a shared or closer understanding about the meaning of a vague spatial concept. If vagueness still exists, they can further reduce it through other collaborative dialogues, e. g. negotiation of the meaning of that concept or direction communication of the meaning of that concept.

The human emulation design principle for the collaborative dialogue approach corresponds to the Human Emulation (HE) approach for design of human-computer collaboration defined in Terveen (1995). This approach views the human user as "rational agents who form and execute plans for achieving their goals and infer other people's plans" and has comparable abilities with the human. The Human Complementary (HC) approach is another approach for design of human-computer collaboration (Terveen, 1995). This approach assumes that the computer and the human user have different abilities and that the computer collaborates with the user by taking on unique responsibilities to complement the user(s) and being a "more intelligent partner". The system designed with the HE approach has comparable abilities with the human and can have more natural communication with the user than that with the HC approach. However, the system with the HE approach needs more domain knowledge and more complicated reasoning than that of the HC approach in order to have comparable abilities with the human.

The plan-based computational representation of the system's knowledge in the *PlanGraph* model related to inferring meanings of vague spatial concepts constrains types of vague spatial concepts that can be communicated between the user and the system with this model. In the current design of the *PlanGraph* model, such system knowledge is encoded as actions and their recipes (Figure **4-9**). Contextual factors considered by the system are modeled as action nodes or parameter nodes in these recipes. The meaning of a vague spatial concept under constraints of these contextual factors is assigned as a parameter value needed in a GIS command related action. Therefore, the *PlanGraph* model works for only vague spatial concepts whose contextual influences can be modeled as action/parameter nodes needed in this model and whose meanings can be converted into a parameter value needed in GIS commands. For example, contextual factors that influence the meaning of the vague spatial concept, *near* (Figure **4-9** (b)), can all be modeled as action/parameter nodes, and, the meaning of *near* can be converted to an absolution spatial distance needed in the GIS command *Spatial Query by Distance*. Thus, the *PlanGraph* model can work for this concept. Another example vague spatial concept for which this model can work is *Overlay* (Zhan, 2002) in case we know how various contextual factors can influence the human agent's adjustment on the *overlay* relation between two regions and they can be represented as action/parameter nodes. This concept is represented by the numeric value of the area-based ratio between two spatial regions in results of the fuzzy membership functions for this concept developed by Zhan (2002). The GIS can understand the numeric value directly.

We need to further improve the generalizability of the *PlanGraph* model. In this chapter, we describe general ways to design actions and recipes in the *PlanGraph* model for the GIS to handle the vagueness problem. By following the general principles of designing such actions and recipes, we can extend the system's knowledge of actions and recipes by following corresponding human knowledge involved in communication of various specific vague concepts in various specific contexts. For example, by following the general recipe in Figure **4-9** (a) for the system to understand the meaning of a vague spatial concept, we can design a recipe in Figure **4-9** (b) for the system to infer the meaning of *near* based on human knowledge of the context-dependent model of the meaning of *near*. If we have knowledge about what and how contextual factors influence the meaning of *overlay*, we can extend the system's knowledge by adding actions and recipes for the system to infer the meaning of *overlay* by following the recipe in Figure **4-9** (a). However, these actions and recipes are not general enough. For example, actions and recipes representing the system's knowledge about the contexts behind the user's spatial information requests are designed differently in different specific contexts, e. g. food shopping, planning for vacation, and looking for jobs (see example actions and recipes in Figure **A-2**, Figure **A-3**, and Figure **A-4**). Each context involves different actions and different parameters. These actions and recipes do not have a general representation that can be applicable in all different contexts. Another example is that parameters representing contextual factors in the general recipe in Figure **4-9** (a) for the system to understand the meaning of a vague spatial concept are also different for different specific vague spatial concepts. These actions and recipes may be further improved so that they do not need to be changed for communication of different concepts in different contexts. For example, the recipe in Figure **4-9** (a) involves fixed parameters representing the same lists of contextual factors which influence meanings of various vague spatial concepts. Further generalization of the *PlanGraph* model can improve the flexibility of this model by easing the application and implementation of this model for various information systems to communicate various vague spatial concepts in a variety of contexts.

Chapter 5

Prototype Software Agent, *GeoDialogue*

A prototype software agent, *GeoDialogue*, is implemented in this study as a dialogue manager for a spoken natural language enabled GIS. It is used to illustrate how the collaborative dialogue approach and the *PlanGraph* model can be implemented for the GIS to manage collaborative human-GIS communication. This chapter describes functions and implementation of this agent in detail.

5.1 Overview

The prototype software agent, *GeoDialogue*, is implemented with a modular architecture as shown in Figure **5-1** (Cai et al., 2003; Cai et al., 2005). It includes five major modules, including *Semantic Analysis*, *Interpretation*, *Dialogue Control*, *Elaboration*, and *Response Control*. In addition, another two modules, *Knowledge Base*, and *GIS Component*, are also included. *Knowledge Base* provides the dialogue agent with general knowledge used in the five major modules. *GIS Component* provides GIS functions and results of GIS commands to the *Elaboration* module when the dialogue agent needs to execute certain GIS commands during the elaboration process.

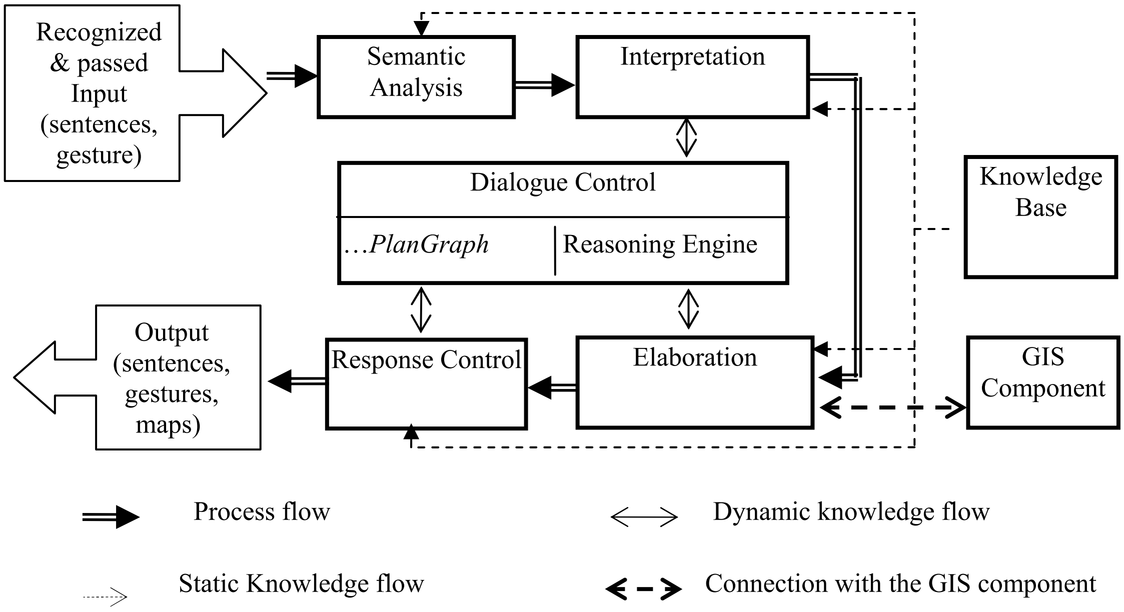


Figure **5-1**: Architecture of *GeoDialogue*

The dialogue agent can take the input in the form of recognized and parsed sentences and/or gestures, and send out the output as mix of sentences, maps, and/or simulated gestures. The input form allows *GeoDialogue* to work with any natural multimodal input devices that combine natural language (spoken or typed) with gestures (pen-based or free-hand). Communications between *GeoDialogue* and input/out devices are through a simple message protocol, DAVE_GXML, which is implemented with common internet message protocols, e. g. XML. An example user input in the format of DAVE_GXML is shown in Figure **5-2**. In this figure, the root element <DAVE_GXML> defines major input components, the utterance and the gesture, which are defined by the child element <UTTERANCE> and <GESTURE>, respectively. The user utterance information is further encoded in child elements, <WORD>, of <UTTERANCE>. Each point in the user gesture is encoded in child elements, <POINTS> and <POINT>, of <GESTURE>.

```
<?xml version="1.0"?>
<DAVE_GXML version="0.0.2">
  <UTTERANCE Speaker="Bob" StartTime="4/15/2006 14:30:15">
    <PHRASE>
      <WORD start_time="82243.12" end_time="82244.2" >zoom</WORD>
      <WORD start_time="82245.22" end_time="82246.4" >to</WORD>
     </PHRASE>
    <PHRASE>
      <WORD start_time="82247.12" end_time="82295.1" >this </WORD>
      <WORD start_time="82293.12" end_time="82294.2" >area</WORD>
    </PHRASE>
   </UTTERANCE>
  <GESTURE StartTime="4/15/2006 14:30:15">
    <POINTS>
      <POINT start_time="82246.4" end_time="82244.2" >
              <X>123</X>
              <Y>245</Y>
      </POINT>
……
      <POINT start_time="82345.4" end_time="82346.2" >
              <X>125</X>
              <Y>248</Y>
      </POINT>
     </POINTS>
   </GESTURE>
</DAVE_GXML>
```

Figure **5-2**: Example User Input to *GeoDialogue*

The remainder of this section is organized as follows. Section 5.2 describes major functions and partial implementation aspects of each module in this agent. Implementation details of the *PlanGraph* model are provided in Section 5.3, which is followed by a summary of this chapter.

5.2 Modules in *GeoDialogue*

Knowledge Base

The *Knowledge Base* module provides general knowledge that the dialogue agent needs to perform the tasks of the other individual modules. This module consists of knowledge about the natural language (and gestures), actions and their recipes, meta data of the GIS spatial datasets, and relations among them. In the current implementation of *GeoDialogue*, the relations among the natural language, actions and GIS datasets are implemented as a relational database, e. g. a SQL database or an ACCESS database. The actual GIS datasets are stored and managed by the GIS component.

Different types of knowledge in the *Knowledge Base* play different roles in the dialogue agent's reasoning and performance in different modules (Figure **5-1**):

- The knowledge about the natural language (the recognized and parsed user input) and its semantic meanings is used to support *GeoDialogue*'s performance in the module of *Semantic Analysis*. It helps the dialogue agent to understand possible semantic meanings of each phrase/gesture in the user input and the overall user input. For example, "I would like to see" in the user request "I would like to see cities near *City A*" means the user's intention to see a map.
- The knowledge about the relation between semantic meanings of the natural language (phrases and/gestures) and certain actions representing the user's intentions/goals supports the dialogue agent's performance in the module of *Interpretation*. For example, based on knowledge about the relation between the semantic meaning of the user request "I would like to see cities near *City A*" and the action *See Map*, the dialogue agent can

interpret the user input by initiating the *PlanGraph* with the action *See Map*.

- The knowledge about the relation between semantic meanings of the natural language (phrases and/gestures) and actions (and their recipes) representing concepts other than the user's intentions is designed to support the agent's performance in the module of *Elaboration*. For example, "cities near City A" in the user request "I would like to see cities near *City A*" means the map content required by the user in the user's intention *See Map*. The knowledge about the relation between the map content and the action *Instantiate Lyr* and its recipe *Select by Distance* is used by the GIS agent to elaborate the parameter *Layer* involved in the plan for the user's intention, *See Map*.
- The knowledge about semantic meanings of natural language (phrases and gestures) and the GIS spatial datasets is used to support the dialogue agent's elaboration process, in particular, in the execution of certain basic actions for determining appropriate geospatial data needed in GIS commands. For example, the semantic meaning of "cities" in the user's request "show me cities" relates to the map layer of cities, as one of the spatial datasets in the GIS spatial database. This knowledge is used by the dialogue agent to instantiate the parameter *Layer* involved in the GIS command, *Show Map*.
- The knowledge about actions and their recipes supports the dialogue agent's elaboration process and response control process. For example, to

instantiate the parameter *Dist* with the action *Instantiate Dist*, the dialogue agent can select the recipe *Infer Near* under constraints of the user input "show me cities near *City A*". Here is another example. In case the recipe *Ask* of *Share Context* is used by the dialogue agent to ask the user for certain context information, with the knowledge from *Knowledge Base*, the dialogue agent knows that the first sub-action *Ask* in this recipe is a communicate action to be executed by itself. So it can send out the response generated by executing *Ask* by applying **ORs**.

GIS Component

The *GIS Component* module provides results to the GIS commands executed in certain actions during the elaboration process in the *Elaboration* module. In the implementation of *GeoDialogue*, the *ArcIMS* is used as a GIS server. It manages spatial datasets and provides basic GIS functions. It accepts GIS command requests, e. g. *Select by Distance*, from the *Elaboration* module and returns responses, e. g. query results and/or map responses, to the *Elaboration* module. The communication between the ArcIMS server and the *Elaboration* module is through the ArcXML language (ESRI, 2002).

Semantic Analysis

The *Semantic Analysis* module starts to understand the meaning of the user input after receiving it (usually, as recognized and parsed sentences with gestures if any). This module first assigns a meaning(s) to each phrase in the user input after retrieving the knowledge about natural language from *Knowledge Base*.

The meaning(s) of each phrase can be a concept referring to an action (e. g. the spatial intention about seeing a map), or a concept referring to spatial information (e. g. distribution of cities), or other entities corresponding to a parameter value (e. g. the spatial distance unit, miles), or the user's beliefs about certain facts (e. g. positive comments on the system's map response). In this study, under the assumption that the user input usually is a simple sentence containing one major user intention, the semantic analysis result of this module usually includes a concept referring to an action representing the user's intention underlying the user input and/or other concepts referring to parameter values.

Interpretation

After the semantic analysis with the user input, the *Interpretation* module starts to interpret the user's intention underlying the user input depending on the relation between the user intention and the existing *SharedPlan* between the GIS and the user (represented as the *PlanGraph* in the *Dialogue Control* module). For example, in case the existing *PlanGraph* is empty, the user intention *See Map* underlying the request "show me cities near City A" is used to initiate the *PlanGraph*.

The interpretation process leads to updating of *PlanGraph*, in particular, the part of *PlanGraph* corresponding to the user's intention underlying the user input and other concepts from the semantic analysis result. Such updates can be either the GIS agent's mental states on action/parameter nodes corresponding to these concepts, or parameter values corresponding to these concepts, or change of the attention focus in *PlanGraph*. For example, after interpreting the user's positive feedback "Yes, thanks" to the system's map response to the user request "show me cities near City A", the GIS updates all its mental states on action/parameter nodes involving vagueness from *near* in the *PlanGraph* as success. After interpreting the user's reply "I am going to walk" to the system's question "what's your transportation mode?", the system updates the parameter value of *Vehicle* as "walking", and shifts attention to the next node in the *PlanGraph* that it can participate in.

The dialogue agent's reasoning process in the *Interpretation* module is controlled by the *Reasoning Engine* in the *Dialogue Control* module.

Elaboration

Upon receiving the interpretation result of the user's input, the *Elaboration* module starts to evolve the existing collaborative human-GIS plan toward the success of the user's spatial information request. The collaborative plan is recorded by *PlanGraph* in the *Dialogue Control* module. In this study, the dialogue agent usually tries to achieve success of the plan for the user's spatial information request, not for the whole plan in the *PlanGraph* (including the plan representing the context behind the user's request, later, as the context plan). This is designed under the assumption that the GIS agent believes that the user is responsible for sub plans in the context plan. For example, the existing *PlanGraph* includes the user's context plan about *Plan for Vacation* and the user's spatial intention *See Map* underlying the user request "show me cities near City A". The system usually focuses on advancing *See Map* towards its success and does not try to help the user's sub plans, e. g. *Travel* and *Visit Targets*, involved in *Plan* for *Vacation*.

The output of the *Elaboration* module usually involves results from execution of certain sub plans during the elaboration process or communicative responses generated by the dialogue agent to communicate mutual beliefs. For example, by executing the sub plan, *Generate Map*, for displaying an existing GIS dataset (as a map layer), the dialogue agent generates not only a map response from the GIS component, but also a natural language response: "Here it is"; by executing a collaborative dialogue plan, *Ask* of *Share Context*, for requesting certain context information *X*, the dialogue agent may generate a natural language response: "What is X?".

The dialogue agent's reasoning in the *Elaboration* module is also controlled by the *Reasoning Engine* in the *Dialogue Control* module.

Response Control

The *Response Control* module is to control when to stop the elaboration process and output all responses from the *Elaboration* module. As described at the beginning of this chapter, the output of this module is in the format of DAVE_GXML. Similarly, the dialogue agent's reasoning in the response control process is also controlled through the *Reasoning Engin*e in the *Dialogue Control* module.

Dialogue Control

The *Dialogue Control* module is the central intelligence module in the dialogue agent. It includes two sub components, *PlanGraph* and *Reasoning Engine*. This module plays three major roles through these two components. The first role is to maintain the dynamic knowledge on the shared context involved in human-GIS communication through the *PlanGraph*. The *PlanGraph* helps the dialogue agent to keep track of the current state of human-GIS collaboration underlying the human-GIS communication. The task context, spatial context and human context are kept in the dynamic knowledge of the human-GIS collaboration process. The dynamic knowledge of the shared context provides help for the dialogue agent to make appropriate reasoning during the human-GIS collaboration process through the *Reasoning Engine* component.

The second role is to perform automated reasoning for the three modules, *Interpretation*, *Elaboration* and *Response Control*, through the *Reasoning Engine* component. This component incorporates the reasoning algorithms, **IAs**, **EAAs** & **EAPs**, and **ORs**, described in Chapter 4. These reasoning algorithms are used to control the dialogue agent's reasoning in *Interpretation*, *Elaboration*, and *Response Control*, respectively.

The third role is to control the dialogue process flow in the dialogue agent through the *Reasoning Engine* in the order shown in Figure **5-1**. Firstly, the user input from the input device directly goes into *Semantic Analysis*. Then, the result of the semantic analysis comes to the *Interpretation* module, followed by the recursive elaboration process. Finally, the responses generated from *Elaboration* are sent out by the *Response Control* module to the output device. The dialogue agent waits for the next user input after outputting responses. In the current version of *GeoDialogue*, the five major modules are implemented as separate functions. The *Reasoning Engine* is implemented as the major function that organizes the information flow among all the other four functions. As in the dialogue process described above, the five functions are organized in *Reasoning Engine* in this order: *Semantic Analysis*, *Interpretation*, *Elaboration*, and *Response Control*.

For implementation details of *PlanGraph*, and reasoning algorithms, see the next section in this chapter.

5.3 Implementation of the *PlanGraph* model

This section describes implementation details of the *PlanGraph* model, including *PlanGraph* and the associated reasoning algorithms.

5.3.1 Implementation of the *PlanGraph*

Knowledge of Actions and Recipes

The basic actions are implemented as public functions in *GeoDialogue*. Complex actions are elaborated and executed after decomposing their recipes and executing basic actions involved in these complex actions. The functions of basic actions can be called when needed during the plan execution process.

The knowledge about recipes of complex actions is stored in *Knowledge Base*. The recipes are implemented with DAVE_GXML. An example recipe in DAVE_GXML is shown in Figure **5-3**. This recipe is the one shown in Figure **4-8** (b) for the GIS agent's action *Show Map*. In Figure **5-3**, attributes of the element <RECIPE> define the recipe's name as "ShowMap", its action name as "ShowMap", and the description of this recipe.

Child elements of <RECIPE> including <Constraints>, <Parameters>, and <Subactions> further describe three major components in the recipe, including constraints, parameters, and subactions. Each individual constraint, parameter, and subaction can be further defined in child elements of these three elements.

```
<?xml version="1.0"?>
<DAVE_GXML version="0.0.1">
<RECIPE Name="ShowMap" ActName="ShowMap" Description="display a map containing a
layer or layers within an extent">
<Constraints Order="false">
</Constraints>
<Parameters Order="false">
  <PARA name="layers" multiple="yes" description= "map layers" optional="true">
     <DESCRIPTION>
         <TYPE type="Layer"/>
     </DESCRIPTION>
  </PARA>
  <PARA name="Extent" multiple="no" description= "map extent" optional="true">
     <DESCRIPTION>
       <TYPE type="envelop, shape, polygon, feature, value"/>
     </DESCRIPTION>
  </PARA>
</Parameters>
<Subactions order="true">
     <SUBACT act="Refresh_Map" Type="Basic"/>
</ Subactions>
</RECIPE>
</DAVE  GXML>
```

Figure **5-3**: Example Recipe Implemented in Knowledge Base

Mental States

A set of integer numbers, Mental State Numbers (MSN), is implemented to label the GIS agent’s different mental states on each action/parameter node in the *PlanGraph* at each major step of the human-GIS collaborative plan evolution process. By referring to general mental states designed in Table **4-1**, Table **4-2**, we use MSNs of 0, 1, 2, 3, 5 and

6 to represent the GIS agent's mental states involved in communication not involving vagueness (Table **5-1** and Table **5-2**).

Table **5-1**: Mental State Number (MSN) on Focus Action α

MSN	Step No of Mental States in Table **4-1**
0	1
1	2
2	3.1
3	3.2.1
5	4 (in case that execution of α fails)
6	4 (in case that execution of α succeeds)

Table **5-2**: Mental State Number (MSN) on Focus Parameter *P*

MSN		Step No of Mental States in Table **4-2**
0		1
1		2
2		3 (in case that p is optional)
3		3 (in case that p is required)
5		4 (in case that instantiation of *P* fails)
6	6_1	4 (in case that instantiation of *P* succeeds for sure)
	6_2	4 (in case that instantiation of *P* succeeds with estimation)

With extensions of mental states involved in handling the vagueness problem, we can further extend the set of MSNs. For example, by referring to the mental states extended in Table **4-3** and Table **4-4**, we can further extend MSNs in Table **5-3** and Table **5-4**.

Table **5-3**: Mental State Number (MSN) on Action α

MSN		Step No of Mental States in Table **4-3**
4	4_2a	4.3 (in case that the GIS agent believes that it needs to revisit α)
	4_2b	4.3 (when the GIS agent right completes α)
	4_3a	4.4 (in case that the GIS agent believes that it needs to revisit α)
	4_3b	4.4 (when the GIS agent right completes α)
	4_4a	4.5 (in case that the GIS agent believes that it needs to revisit α)
	4_4b	4.5 (when the GIS agent right completes α)
5		4.1 (in case that execution of α fails)
6		4.2 (in case that execution of α succeeds)

Table **5-4**: Mental State Number (MSN) on Parameter *P*

MSN		Step No of Mental States in Table **4-4**
4	4_2a	4.3 (in case that the GIS agent believes that it needs to revisit *P*)
	4_2b	4.3 (when the GIS agent right completes *P*)
	4_3a	4.4 (in case that the GIS agent believes that it needs to revisit *P*)
	4_3b	4.4 (when the GIS agent right completes *P*)
	4_4a	4.5 (in case that the GIS agent believes that it needs to revisit *P*)
	4_4b	4.5 (when the GIS agent right completes *P*)
5		4.1 (in case that execution of α fails)
6		4.2 (in case that execution of α succeeds)

For all MSNs and their meanings implemented in *GeoDialogue*, refer to Appendix B.

PlanGraph

PlanGraph is implemented as a recursive dynamic data structure in *GeoDialogue* (Figure **5-4**). The recursive structure supports the dialogue agent's recursive reasoning involved in construction of the *SharedPlan* built by the GIS and the user. This data structure defines the root plan and current attention focus. The current focus in the *PlanGraph* can be an action node or a parameter node) in *PlanGraph*.

```
Public root As clsPlan              ' Root plan of PlanGraph
Public Focus As Object              ' Attention focus in PlanGraph, which can be a plan class
                                    ' (clsPlan) or a parameter class (clsPara)
Private Sub Class_Initialize()
   Initialize
End Sub

Public Sub Initialize()
   Set root = Nothing
   Set Focus = Nothing
End Sub
```

Note: The meaning of each variable is explained in the comment after definition of the variable.

Figure **5-4**: Class of *PlanGraph* in Visual Basic Language

The recursive tree structure of *PlanGraph* is achieved through the recursive tree structure of the plan for an action and that of the parameter node. The recursive data structure of the plan for an action (Figure **5-5**) defines all properties of the plan, including the parent object (which can be a plan or a parameter), subplans of the plan, the GIS agent's mental state on this plan, and other properties. Similarly, the data structure of the parameter node (Figure **5-6**) also defines the properties of the parameter, including its parent object (which can be a plan only in the current implementation), its subplans (which represents all plans used to instantiate this parameter), its parameter values that have been instantiated so far, the GIS agent's mental state on this parameter and other properties.

```
Public Action As String                  ' Action name
Public ActionType As String              ' Complexity of the action: Complex or basic
Public parent As Object                  ' Parent object can be either clsPara or clsPlan
Public MSN As enumPlanMentalState        ' GIS's Mental State Number on the plan
Public Recipes As Collection             ' All recipes for the same action
Public CurrentRecipe as XML              ' The current recipe used
Public Parameters As Collection          ' All parameters (type is clsPara)
Public Agents As Collection              ' All participating agents
Public Subplans As Collection            ' plan collection

Private Sub Class_Initialize()
   Set Subplans = New Collection
   Set Parameters = New Collection
   MSN = enumMentalStateNumber.0
End Sub

Private Sub Class_Terminate()
   Set Subplans = Nothing
   Set Parameters = Nothing
End Sub
```

Note: The meaning of each variable is explained in the comment after definition of the variable.

Figure **5-5**: Class of a Plan in Visual Basic Language

```
Public name As String                    ' Parameter name
Public parent As clsPlan                 ' Parent, a plan object, of this parameter
Public Multiple As Boolean               ' if this parameter can have multiple values or not
Public paraType As String                ' parameter type: Layer, shape, point, line, polygon,
                                         ' extent, integer, real

Public ParaDescription As String         'Description of this parameter
Public Optionalness As Boolean           'If this parameter is optional or not
Public MSN As enumParaMentalState        ' GIS's Mental State Number on this parameter
Public paraList As Collection            'All values instantiated so far for this parameter
Public Subplans As Collection            'Collection of plans to instantiate this parameter

Private Sub Class_Initialize()
   Set Subplans = New Collection
   MSN = enumParaMentalState.0
   Set paraList = New Collection
End Sub

Private Sub Class_Terminate()
   Set paraList = Nothing
   Set Subplans = Nothing
End Sub
```

Note: The meaning of each variable is explained in the comment after definition of the variable.

Figure **5-6**: Class of a Parameter in Visual Basic Language

5.3.2 Implementation of Reasoning Algorithms

All the reasoning algorithms associated with the *PlanGraph* are implemented as separate functions in *GeoDialogue*. For convenience of describing different reasoning algorithms, we label these algorithms with letters and numbers. For all reasoning algorithms with such labels implemented in *GeoDialogue*, see Appendix C. Concerning to reasoning algorithms which are designed for the GIS agent to process human-GIS communication without vagueness, we have implemented them by referring to MSNs in Table **5-1** and Table **5-2**. In case the GIS agent does not need to handle the vagueness problem, the dialogue agent can use reasoning algorithms in this category, e. g. **IAs** including **IA1**,

IA3, and **IA4**, **ORs** including **OR1** and **OR2**, **EAAs** including **EAA0**, **EAA1**, **EAA2**, **EAA3(a, b1)**, **EAA5**, **EAA6, EAAU and EAAC**, and **EAPs** including **EAP0**, **EAP1**, **EAP2/3 (c,d)**, **EAP4_1**, **EAP5**, and **EAP6**. Regarding **EAAs** and **EAPs** which are specifically designed for the GIS agent to handle the vagueness problem in human-GIS communication, we have implemented them with referring to MSNs in Table **5-3** and Table **5-4**, including **EAA4_2**, **EAA4_3**, **EAA4_4**, **EAP4_2**, **EAP4_3**, and **EAP4_4**, **EAP1(a)** and part of **EAP1(b)**.

The *Reasoning Engine* component organizes these reasoning algorithm functions into different module functions to support the dialogue agent's automatic reasoning in the modules of *Interpretation*, *Elaboration* and *Response Control*. The three module functions responsible for these three modules can access all **IA** functions, all **OR** functions, and all **EAA** and **EAP** functions, respectively. The two module functions for *Interpretation* and *Response Control* are relatively simpler compared with the function responsible for the module of *Elaboration*. The module function for *Elaboration* (Figure **5-7**) integrates functions of **EAAs** and **EAPs** in the recursive way because the dialogue agent's elaboration process is a recursive process. The dialogue agent's elaboration reasoning always starts by calling the function, the procedure *Elaborate* (*PlanGraph*) in Figure **5-7**, to elaborate the current focus of the *PlanGraph*. This is implemented by following the elaboration principle that the GIS agent's elaboration process always starts with elaboration of the current focus on the *PlanGraph*. If the current focus is a plan for an action, the process of *Elaborate* (*PlanGraph*) will call the procedure, *Elaborate_Plan* (*Plan for an action*) in Figure **5-7**, to elaborate the focus plan. If the current focus is a parameter node, the procedure, *Elaborate_Parameter*

(*Parameter*) in Figure **5-7**, will be called. The result of elaborating a plan or a parameter will lead to updating of the current attention focus and mental states on the plan/parameter node on the *PlanGraph*. After applying each **EAA** or **EAP** function, the elaboration process will continue to elaborate the *PlanGraph* by calling the function *Elaborate* (*PlanGraph*). Thus this function will continue to elaborate the updated attention focus in the *PlanGraph* until there is a need to stop the elaboration process.

```
Sub Elaborate (PlanGraph) ' Start elaboration with the current attention focus in PlanGraph
    If Focus in PlanGraph is an action, then
        Elaborate_Plan (Focus);
    Else
        Elaborate_Parameter (Focus);
    End if
End Sub

Sub Elaborate_Plan (Plan for α) ' All elaboration algorithms are implemented here.
    Select Case MSN on α
        … …
        Case i
            'Reasoning algorithm for elaborating α is implemented in the function
            'below
            EAAi (Plan for α)
        … …
    End Select

    'After applying the reasoning algorithm, MSN on α and attention focus in
    'PlanGraph can be changed. So, the GIS continues to elaborate
    'PlanGraph recursively
    Elaborate (PlanGraph)

End Sub

Sub Elaborate_Parameter (Parameter P) 'All elaboration algorithms are implemented here.
    Select Case MSN on P
        … …
        Case i
            'Reasoning algorithm for elaborating P is implemented in the function
            'below
            EAPi (Parameter P);
        … …
    End Select

    'After applying the reasoning algorithm above, MSN on P and attention
    ' focus in PlanGraph can be changed. So, the GIS continues to elaborate
    'PlanGraph recursively
    Elaborate (PlanGraph)

End Sub
```

Note: The meaning of each variable is explained in the comment after definition of the variable.

Figure **5-7**: Pseudo Code of the Recursive Elaboration Process

5.4 Summary and Discussion

Details of the prototype software agent, *GeoDialogue*, are described in this chapter. This agent incorporates the collaborative dialogue approach and the *PlanGraph* model to keep knowledge on the dynamic human-GIS communication process and reason about how to react to each of the user's utterances collaboratively. *GeoDialogue* has a modular architecture. Its core module, *Dialogue Control*, incorporates the *PlanGraph* model to maintain dynamic knowledge and perform automatic reasoning during human-GIS communication. One of the two key components of the *PlanGraph* model, *PlanGraph*, is implemented in *Dialogue Control* as a recursive dynamic data structure, which supports the dialogue agent's recursive reasoning involved in the construction of the *SharedPlan* built by the GIS and the user. The other key component of the *PlanGraph* model, associated reasoning algorithms, is implemented as separate functions in the *Reasoning Engine* component of *Dialogue Control*. *Reasoning Engine* further organizes these reasoning algorithm functions into different module functions to support the dialogue agent's automatic reasoning in the other major modules in this agent.

The knowledge built in the *PlanGraph* model and *GeoDialogue* can be further improved to be dynamically updated through incorporation of machine learning methods (Jebara, 2004) in this model and the dialogue agent. In the current implementation of the *PlanGraph* model in *GeoDialogue*, the system's spatial knowledge about meanings of specific vague spatial concepts in various specific contexts is pre-built in the system's knowledge base. The pre-built knowledge does not change like human knowledge acquisition as human-GIS communication experiences increase. In the future, we can incorporate machine learning approaches in design and implementation of the *PlanGraph* model and the dialogue agent so that the dialogue agent can dynamically accumulate and update the spatial knowledge related to context-dependent meanings of vague spatial concepts based on its experiences of human-GIS communication of vague spatial concepts in various contexts. The dynamic learning capability provided by machine learning approaches would ease implementation of this model and the dialogue agent. The system with dynamically accumulated and updated spatial knowledge would be able to communicate with the user more intelligently like a human agent by adjusting its inference on the meaning of a vague spatial concept based on its previous communication experience in a context similar to the current context.

Implementation of the dialogue agent, *GeoDialogue*, demonstrates that the collaborative dialogue approach and the *PlanGraph* model can be used for the spoken natural language enabled GIS to manage human-GIS communication and handle the context-dependency problem. They can help the system to react to each of the user utterances appropriately. For example, they support the system to interpret the user's input under constraints of the current communication, help the user by selecting and executing appropriate GIS commands, and return appropriate responses to the user. They support the system to handle the context-dependency problem like a human. For example, they support the system to keep track of the dynamic human-GIS communication, build a shared context with the user through collaborative dialogues, and reach a shared understanding about the vague spatial concept with use of shared context information.

Implementation of *GeoDialogue* also demonstrates that the collaborative dialogue approach and the *PlanGraph* model can help to bridge the gap between the advanced natural multimodal interfaces for GIS and the insufficient human-GIS interaction style. Much progress has been made on the development of natural modalities for the geographic information use, e. g. CUBISON (Neal et al., 1989), *QuickSet* (Cohen et al., 1989; Oviatt, 1992, 1996) and *Sketch and Talk* (Egenhofer, 1996). However, the human-GIS interaction style in these systems is still the standard interaction style in WIMP-based GISs (question-and-answer style), which is a user-initiative interaction style. This interaction style is not sufficient for the natural multimodal GIS, in particular, the conversational GIS to communicate with the user naturally (Egenhofer & Kuhn, 1999). Implementation of *GeoDialogue* demonstrates that the collaborative dialogue approach and the *PlanGraph* model support the system to interact with the user like a human agent. *GeoDialogue*-user interaction is mix-initiative like human-human communication. The user can ask *GeoDialogue* for spatial information or other questions and the system can also initiate dialogues to ask the user for contextual information or other questions if needed. Therefore, the collaborative dialogue approach and the *PlanGraph* model can help to improve the human-GIS interaction for conversational GIS.

Success of the *PlanGraph* model in *GeoDialogue* also shows the possibility that the collaborative dialogue approach and the *PlanGraph* model could be extended to support human-GIS dialogue management for other natural multimodal GIS, e. g. eye-track enabled, and/or facial gesture enabled GIS, as well as speech/written natural language and free-hand gesture/pen-based gesture enabled GIS. They could be extended to help the system to understand such new natural modal user input based on its relevance to the current communication, and then decide how to help the user to reach the user's goal. Therefore, they could help to advance development of natural multimodal GIS.

Chapter 6

Handling the Context Dependency Problem in Various Situations

The GIS and the user can have different understandings of the current context in various situations during the human-GIS communication process. Therefore, the system needs to have flexibility to handle the context-dependency problem in various situations. This chapter provides classification and analysis of these various situations first, and then describes several sample human-*GeoDialogue* dialogues. These sample dialogues illustrate how the collaborative dialogue approach and the *PlanGraph* model can enable the GIS agent to have such flexibility.

6.1 Various Situations Involving Vagueness in Human-GIS Communication

As described in Chapter 3, human communicators may have different understandings about the current context in different situations. In addition, the human communicators may have different models about the meaning of a vague spatial concept (see Eq. **3.1**). Similarly, human-GIS communication can also involve vagueness in various situations of different understandings about the context and different models of the meaning of the vague spatial concept.

Similar to human communicators, the GIS agent and the user can also have different understandings about the context involved in communication in various situations. Concepts referring to contexts (simply as "context concepts" later) in the human user's conceptual system are represented as frames and associated perceptual symbols with recursive tree structures (see Chapter 3). In this study, the GIS agent incorporating the *PlanGraph* model also holds knowledge about the context concepts with recursive tree structures represented in *PlanGraph* (see Chapter 4). The instantiations of the context between the GIS agent and the human user can be different in various situations (Table **6-1**). In each of the situations described in Table **6-1**, the GIS agent and the user can have different models of the meaning of the vague spatial concept, even if they reach a shared understanding about the context. This section describes these various situations of the context-dependency problem in human-GIS communication.

Table **6-1**: Various Situations of the Context Dependency Problem in Human-GIS Communication

Situation Category	Description
A	The GIS agent and the human user have different understandings about the context at the top level of the context concept.
B	The GIS agent and the human user understand the context with the same recursive structure, but with different instantiation values of part of its components.
C	The GIS agent and the human user understand the context with different depths of the recursive structure.
D	The GIS agent and the human user understand the context with different widths of the recursive structure.
E	The GIS agent and the human user understand the context with different widths, depths, and instantiation values of part of components of the recursive structure.

6.1.1 Situation A: Understanding Context with Different Concepts

In **Situation A**, the GIS agent and the human user have different understandings about the current context involved in communication from the top level of the context concept structure (Figure **6-1**). In this situation, the GIS agent understands the current context behind the user’s spatial information request with a concept totally different from what the human user intends. For example, the user’s final goal in the context behind the user’s request “show me cities near City A” is for the user to look for jobs in the cities near City A, while the GIS agent infers the context behind the user’s request as planning for vacation. In this example, the GIS agent’s inference about the context differs from the user’s understanding about the context from the root of the recursive structure of the context concept.

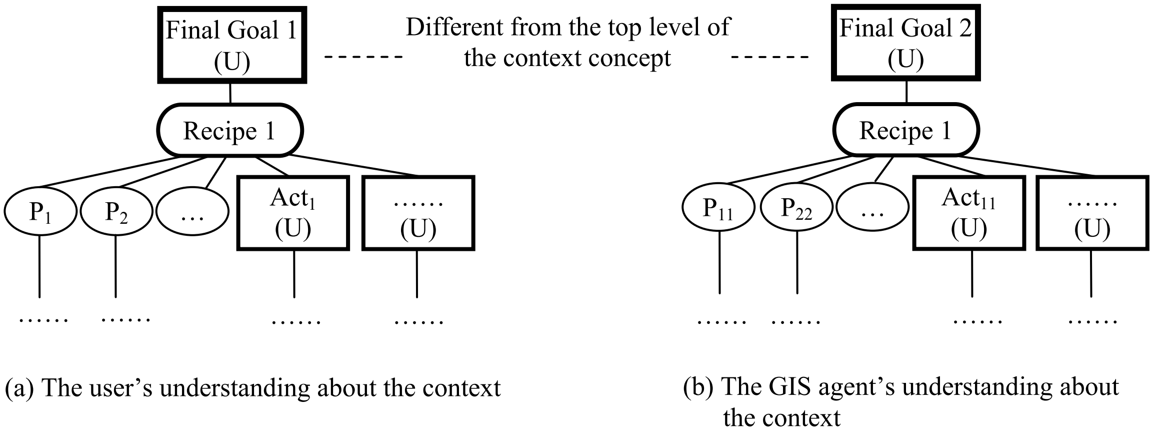
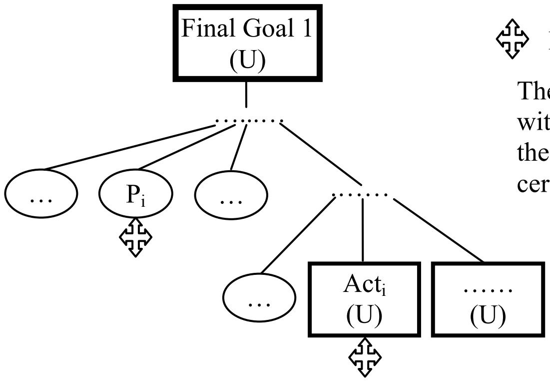


Figure **6-1**: Context-Dependency Problem in **Situation A**

In this situation, even if the GIS agent holds complete knowledge about the human context including the user’s model about the meaning of the vague spatial concept in various contexts, meanings of the vague spatial concept instantiated in the GIS agent and the user are usually different.

6.1.2 Situation B: Understanding Context with Different Sub-Component Values

In **Situation B**, the GIS agent and the human user understand the context as the same context concept with the same recursive structure, but with different instantiations of part of the components in the structure (Figure **6-2**). Thus, in this situation, in the two agents's understandings of the current context, the recursive structures of the context concept are instantiated with same width and same depth, and the top level component of the recursive structures are instantiated with the same value. However, certain sub components of the context structure are instantiated with different values. In this situation, the context factors about which the GIS agent and the user have different understandings can be any task, spatial or human contextual factors, or any combination of them. If these context factors play influential roles on the meaning of the vague spatial concept, meanings of the vague spatial concept instantiated by the GIS agent and the user usually will be different.



✥ Different instantiation

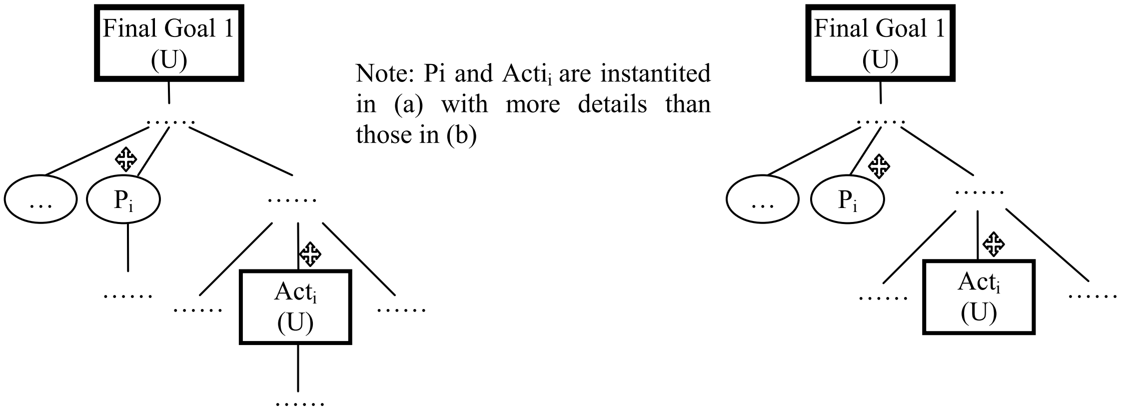
The GIS and the user mentally simulate the context with the same structure of the context concept, but their instantiations of the context concept differ at certain component values, e. g. P_i and Act_i.

Figure **6-2**: Context-Dependency Problem in **Situation B**

Here is an example of the concept dependency problem in this situation. As to the context behind the human user's request "show me grocery stores near Apartment A", both the user and the GIS agent understand the context as planning for food shopping in grocery stores in the urban area around Apartment A. This context involves a traveling activity and a shopping activity. The GIS agent infers the transportation mode involved in the traveling activity as driving a car, while the user actually plans to walk. The GIS agent knows that the spatial area around Apartment A is a metropolitan area from its spatial database, while the user considers it as a small city because he/she is not familiar with it. So, in this example, the context meant by the user is the same as that inferred by the GIS agent at the top level of the context structure, but different regarding the instantiation of the transportation mode and the spatial area in the context. In this example, both the GIS agent and the user consider the transportation mode and the spatial size of the area as important context factors that influence the meaning of *near*. Thus, before they reach a shared understanding about the context, their understandings about *near* will be different.

6.1.3 Situation C: Understanding Context with Different Depths

In **Situation C**, the GIS agent and the human user understand the context with the same context concept, but with different depths of the recursive structure (Figure **6-3**). In the example in Figure **6-3**, the two agents' understandings about the context are the same at the top level of the recursive structure, but different at depths of the component P_i and Act_i. These components can correspond to various context factors. In this situation, usually, the GIS and the user consider the influence of different context factors and have different models of the meaning of the same vague spatial concept.



Note: Pi and $Acti_i$ are instantited in (a) with more details than those in (b)

(a) The user's understanding about the context

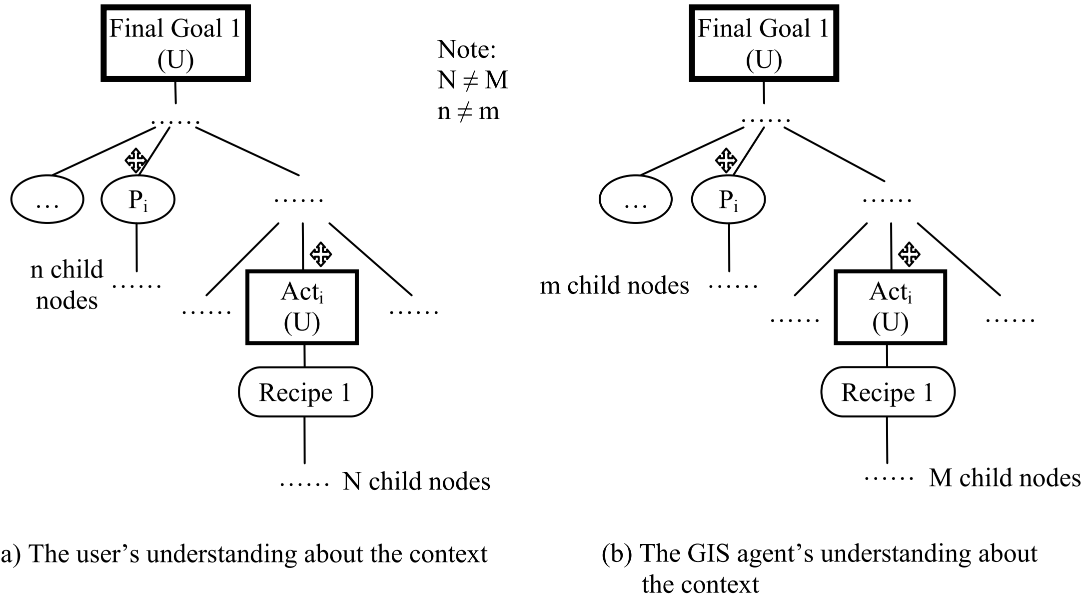
(b) The GIS agent's understanding about the context

Figure **6-3**: Context-Dependency Problem in **Situation C**

Here is an example of the context-dependency problem in **Situation C**. In human-GIS communication about the human user's request "show me grocery stores near Apartment A", both the user and the GIS agent's understand the context of planning for food shopping in grocery stores in the urban area around Apartment A. The GIS agent does not instantiate the *Buy Food* activity involved in the context with any more details, while the user does it with more details of *Food*, e. g. food prices and food qualities, and more details of the *Buy* action, e. g. acceptable payment methods, the casher's service quality, etc. Similarly, the user also considers the routes involved in the traveling activity in the context with more details than the GIS agent does, e. g. their spatial distribution, connectivity, speed limits, etc. In this example, the user considers more contextual influence from details of the shopping activity and the routes on the meaning of *near* than the GIS agent does.

6.1.4 Situation D: Understanding Context with Different Widths

In **Situation D**, the GIS agent and the human user understand the context with the same context concept, but with different widths of the recursive structure (Figure **6-4**). So, in this situation, the two agents' understandings about the current context are the same at the top level of the context concept, but different regarding widths of part of components (e. g. P_i and Act_i shown in Figure **6-4**) of the context structure. Similar to **Situation C**, in this situation, these components can correspond to various context factors. The GIS and the user usually consider influences of different contextual factors and have different models of the meaning of the same vague spatial concept.



(a) The user's understanding about the context

(b) The GIS agent's understanding about the context

Figure **6-4**: Context-Dependency Problem in **Situation D**

Here is an example of the context-dependency problem in **Situation D**. In communication about the human user's request "show me grocery stores near Apartment A", both the user and the GIS agent understand the context as planning for food shopping in grocery stores in the urban area around Apartment A. The user instantiates the food shopping context as an event involving not only the shopping activity and the traveling activity, which are agreed on by the GIS agent, but also the frequency of the traveling and shopping activities. The user understands the *Buy Food* activity involved in the context with more details than the GIS agent, including not only the food and the shopping action, but also other services associated with this activity, e. g. parking spaces, return services, etc. The user instantiates the routes involved the traveling activity involved in the context with its traffic status, as well as its spatial distribution. In this example, the user considers contextual influence from the shopping frequency, the traffic status of the route, and associated services in the *Buy Food* activity for the meaning of *near*, in addition to the contextual influence considered by the GIS agent.

6.1.5 Situation E: Mix of Context-Dependency Problems

In **Situation E**, the GIS agent and the human user understand the context as the same context concept, but with different details of the recursive structure (Figure **6-4**). Thus, in this situation, the two agents' understandings about the current context are same at the top level of the context concept. However, they understand part of components (e. g. P_i and Act_i shown in Figure **6-4**) of the context structure with different instantiation values and/or different widths and/or different depths. These components can correspond to various context factors. The context-dependency problem in this situation may involve a mix of the context-dependency problems in **Situation B**, **C**, and **D**.

6.1.6 Summary and Discussion

This sub-section describes five categories of various situations involving the context-dependency problem in human-GIS communication. The GIS agent and the user can understand the current context behind the user's spatial information request in different ways in different categories of situations. In each category of such situations, the two agents' different understandings about various contextual factors lead to various situations of the context-dependency problems.

The remainder of this chapter describes several sample human-GIS dialogues involving the context-dependency problem in various situations. The GIS agent is *GeoDialogue*. The user agent may vary in different dialogues. For convenience of description, we suppose the user agent to be a male. In the GIScience community, the concept *near* referring to the spatial distance relation, as a typical example of a vague spatial concept, has been investigated for many years, in particular, for its context-dependency problem. In these dialogues, we also use this concept to illustrate how the GIS agent can handle the context-dependency problem with the collaborative dialogue approach and the *PlanGraph* model.

In the remainder of this chapter, we provide a sample human-GIS dialogue in one situation in each of the first four categories of situations described in this section and explain how the collaborative dialogue approach and the *PlanGraph* model can help the GIS agent to handle the vagueness problem in that situation. The context-dependency problem in **Situation E** may involve various combinations of context-dependency problems in **Situation B**, **C** and **D**. The GIS can handle the context-dependency problem in **Situation E** by integrating reasoning and collaborative dialogues used in these three situations. Thus, we do not present an example of how the GIS can handle the context-dependency problem in **Situation E** in the remainder of this chapter.

The *PlanGraph* model needs to be extended to help the GIS agent to handle the context-dependency problem in different situations in these dialogues. Firstly, the four sample dialogues involve different events that the user will participate in, and different spatial information requests. Thus, the GIS agent needs to have different knowledge to understand such events and such information requests. Secondly, the context-dependency problems in these dialogues are in different situations. The GIS agent needs to use different collaborative dialogue strategies and associated reasoning to handle them in different situations. As a result, we need to further extend the *PlanGraph* model described in Chapter 4 with knowledge of actions and recipes and reasoning algorithms involved in handling the context-dependency problem in each dialogue. For each dialogue, we first explain the extension of the *PlanGraph* model, and then describe the process for the GIS agent to handle the context-dependency problem in each dialogue by using the extended *PlanGraph* model. For details about the actions and recipes, MSNs, and reasoning algorithms addressed in these examples, refer to Appendix A, B and C.

6.2 Scenario 1: Handling Context-Dependency in Situation A

A sample human-*GeoDialogue* dialogue involving the context-dependency problem in **Situation A** is shown in Figure **6-5**. In this scenario, the GIS agent's model of the meaning of the vague spatial concept includes the same list of contextual factors as those in the user's model.

G: GIS (*GeoDialogue*) U: User (User-A)

Before the following conversation, the map on the GIS shows a basic map of the destination area, including only state boundaries and interstates (Figure **6-6**).

U (1): Hi, I am User-A. I want to see cities near City A.
G (1): (highlight certain cities near City A) For the plan of vacation purpose by driving, here are cities within 300 miles of City A. Is this OK?
U (2): No.
G (2): What is your goal in asking for this map?
U (3): I am planning to look for jobs in cities near City A. I plan to work there by driving from City A.
G (3): (Highlight certain cities near City A) Here are cities within 150 miles of City A, is this OK?
U (4): Yes. Thanks.

Figure **6-5**: **Scenario 1**

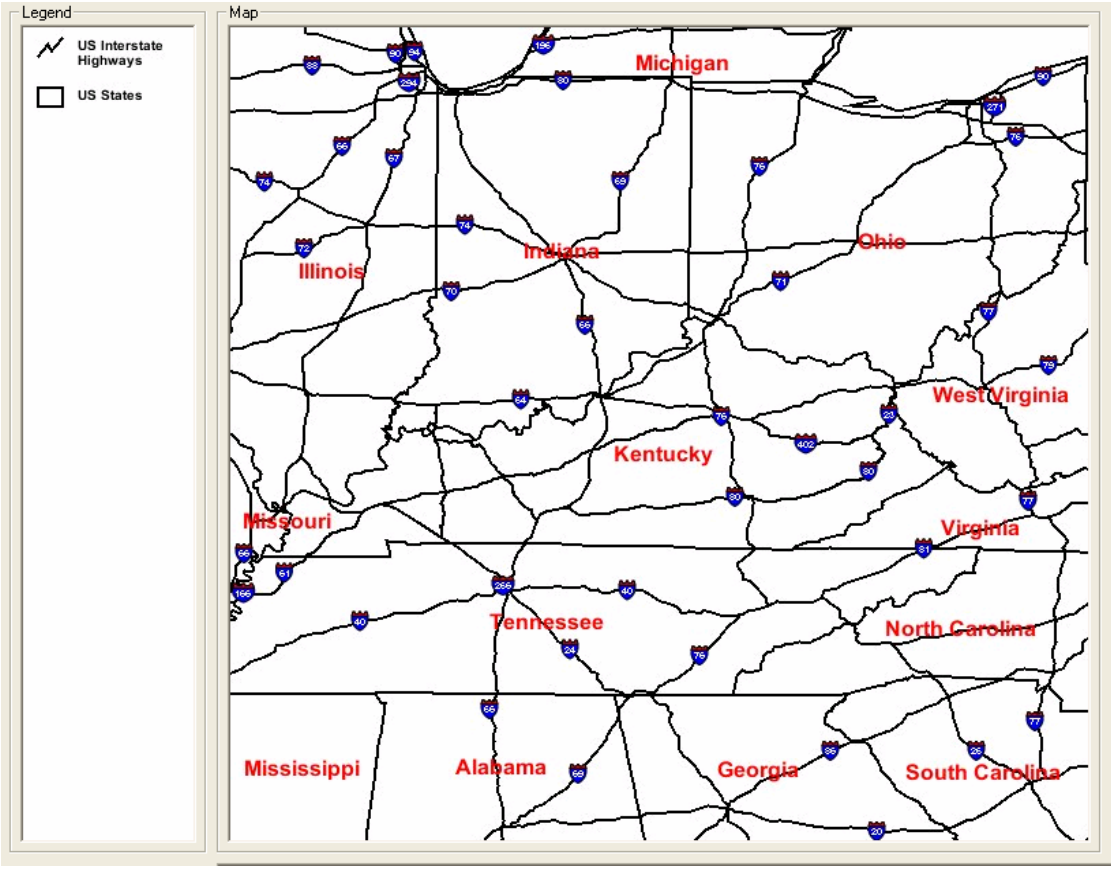


Figure **6-6**: Initial Map in **Scenario 1**

6.2.1 Extension of the *PlanGraph* Model

To enable the GIS agent to handle the vagueness problem in **Scenario 1**, we have implemented a series of actions and recipes by following the general actions and recipes involved in handling the vagueness problem.

To enable the GIS agent to understand the meaning of *near* (involved in the user's spatial information request in U (1)) with context information, we have implemented the recipe *Infer Near* (Figure **A-9** (b)) by following the general recipe *Infer VSC* (Figure **4-9** (a)). The recipe *Infer Near* involves seven optional parameters representing seven contextual factors in the GIS agent's model of the meaning of *near* and a basic action to calculate a spatial distance value corresponding to *near* with context information modeled in these parameters. In this study, the calculation is based on the GIS agent's model of the meaning of *near* in Eq. **6.1**:

$$M_G = f_G(T\_Fg, T\_Tm, S\_Sc, S\_Rt, S\_Rf, S\_Tg, H\_A) \qquad \textbf{6.1}$$

Where: T_Fg, T_Tm, S_Sc, S_Rt, S_Rf, S_Tg, and H_A respectively denote the user's final goal, the transportation mode, the spatial scale/size of area considered, the route distribution, the reference location and the targets involved in the *near* spatial distance relation, and the user name. In the current implementation of *GeoDialogue*, for simplicity, the human agent's name is the only human context factor considered in the GIS model. The GIS agent holds knowledge about some users' understandings about *near* in various contexts with use of the model in Eq. **6.1**.

To enable the GIS agent to infer the context of planning for vacation after receiving U(1) and understand the context of looking for jobs in U (2) behind the user's spatial information request, we have implemented recipes for *Plan for Vacation*, *Visit* and *Travel* in Figure **A-2**, and recipes for *Look for Jobs* and *Work on Jobs* in Figure **A-4**, by following the general actions and recipes in Figure **4-10**. In this scenario, the user's spatial intention *See Map* in U (1) will be attached with the parameter *Destination* in the action *Travel* because the cities as the map content are the destination locations in the user's traveling activity.

To enable the GIS agent to instantiate parameters in the recipe *Infer Near* by inferring/retrieving context information, we have implemented actions and recipes for *Instantiate Goal*, *Instantiate Tm*, *Instantiate Sc*, *Instantiate Rf*, *Instantiate Tg*, and *Instantiate U* by following the general action and recipes in Figure **4-11**. Some of these are shown in Figure **A-10**.

To enable the GIS agent to initiate context-sharing collaborative dialogues between G (2) and the first half of U (3), we have designed and implemented the recipe, *Ask*, in Figure **A-11** (b) for the action *Share Context* (Figure **4-12**). In addition, to enable the GIS agent to understand the user's active context-sharing dialogue in the second half of U (3) "I plan to work there by driving from City A.", we have designed and implemented the recipe, *Actively Give*, in Figure **A-11** (c) for the action *Share Context*.

6.2.2 Process of Handling the Context-Dependency Problem in Situation A

Initial Shared Context before Communication

In this scenario, before U (1), concerning the context to be involved in communication about the vague spatial concept *Near*, *GeoDialogue* and the user agent have shared nothing but partial spatial context through the existing GIS map in Figure **6-6**. The user holds details about the task context and the human context, and the GIS agent has detailed knowledge about the spatial context.

Building Shared Context through U(1) and G (1)

After receiving the user’s request in U (1), by applying general reasoning algorithms for interpretation and elaboration, *GeoDialogue* initiates the *PlanGraph* with the user’s intention *See Map* and recursively elaborates it until it reaches to instantiate the parameter *Dist* in the plan *Select by Distance* with the action *Instantiate Dist*.

In this study, the GIS agent always starts by using the recipe *Infer VSC* to instantiate a parameter from a vague spatial concept by following the HCP1(a). In this scenario, *GeoDialogue* uses the recipe *Infer Near* to assign a spatial distance value to *Dist*. By applying **EAP1(a),** which is specifically designed for retrieving/inferring contextual information, the GIS agent starts to infer the context behind *See Map* before instantiating the optional parameters in *Infer Near* representing contextual factors. Based on the GIS agent’s best knowledge, the GIS agent believes the user’s final goal of *See Map* is to *Plan for Vacation*. Therefore, it integrates this action with the existing *PlanGraph* after elaborating it with actions and recipes designed in Figure **A-2** (a) and (c). Here, *See Map* is attached to the parameter *Destination* in the sub action *Travel* of *Plan for Vacation*. In the context of *Plan for Vacation*, the GIS agent infers the spatial size of the area as the existing map extent area and the transportation mode as driving.

After elaborating *Plan for Vacation*, the GIS agent returns attention back to instantiation of the parameters in *Infer Near*. It successfully instantiates them with the recipe *Look* (Figure **A-10** (b)), which is specifically designed for retrieving contextual information from the existing *PlanGraph*. The *near* distance is inferred as 300 miles based on the GIS agent’s knowledge of User-A’s understanding about *near* in the inferred context.

With the inferred *near* distance, the GIS agent completes the remaining steps in the existing *PlanGraph* for the success of *See Map* and sends out the responses in G (1). During this process, due to the vague spatial concept involved in collaboration, the GIS agent updates its MSNs on *Instantiate Dist* and *Dist* as 4_2b, and MSNs on all their ascendant nodes as 4_4b. By applying **EAA4_2(b)**, **EAA4_4(b)**, **EAP4_2(b)**, and **EAP4_4(b)** specifically designed for handling vagueness, the GIS agent continues the elaboration process with parameters and actions involving vagueness. The current *PlanGraph* after G (1) and the map response in G (1) are shown in Figure **6-7** and Figure **6-8**, respectively.

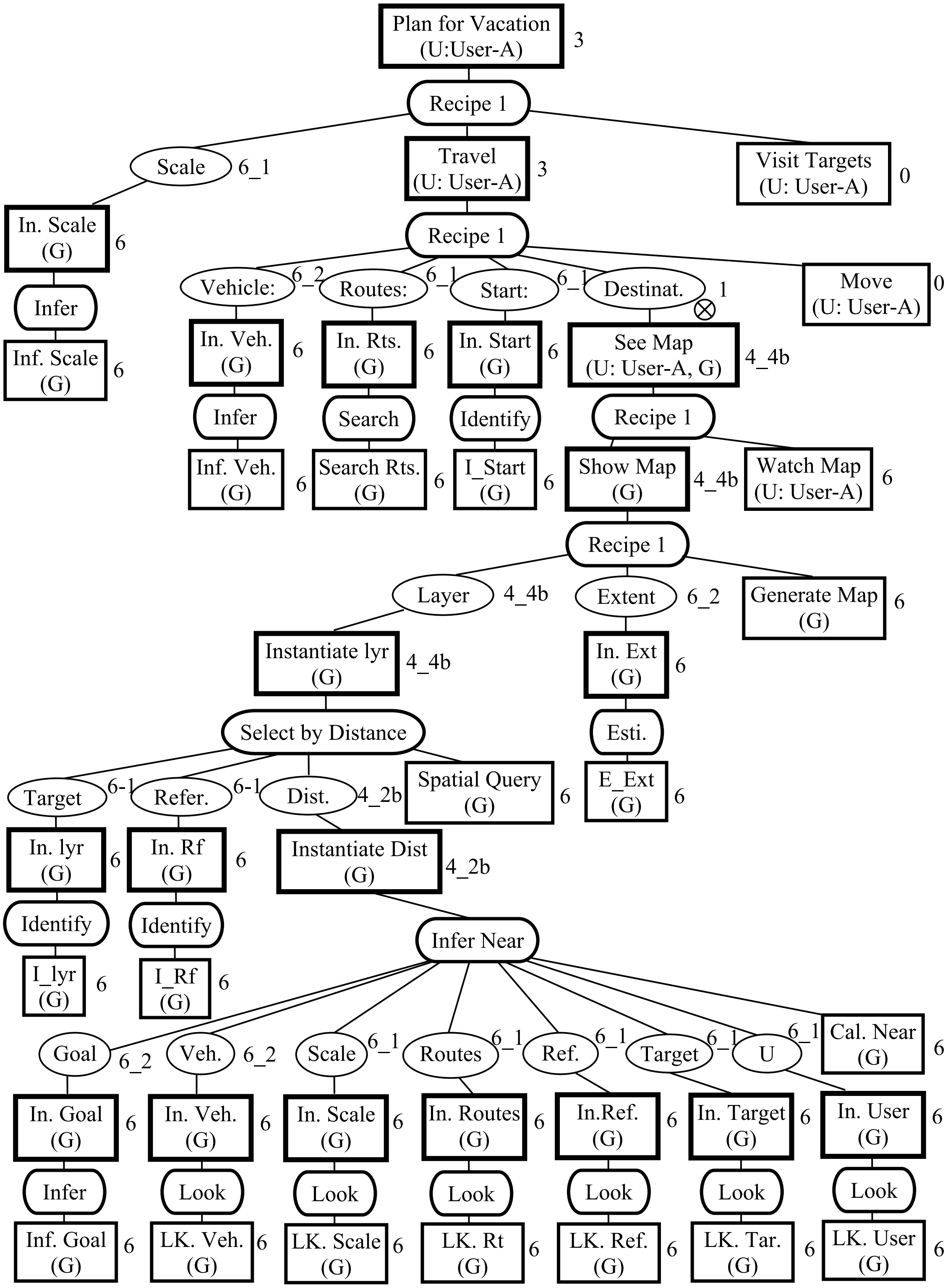


Figure **6-7**: *PlanGraph* after G (1) in **Scenario 1**

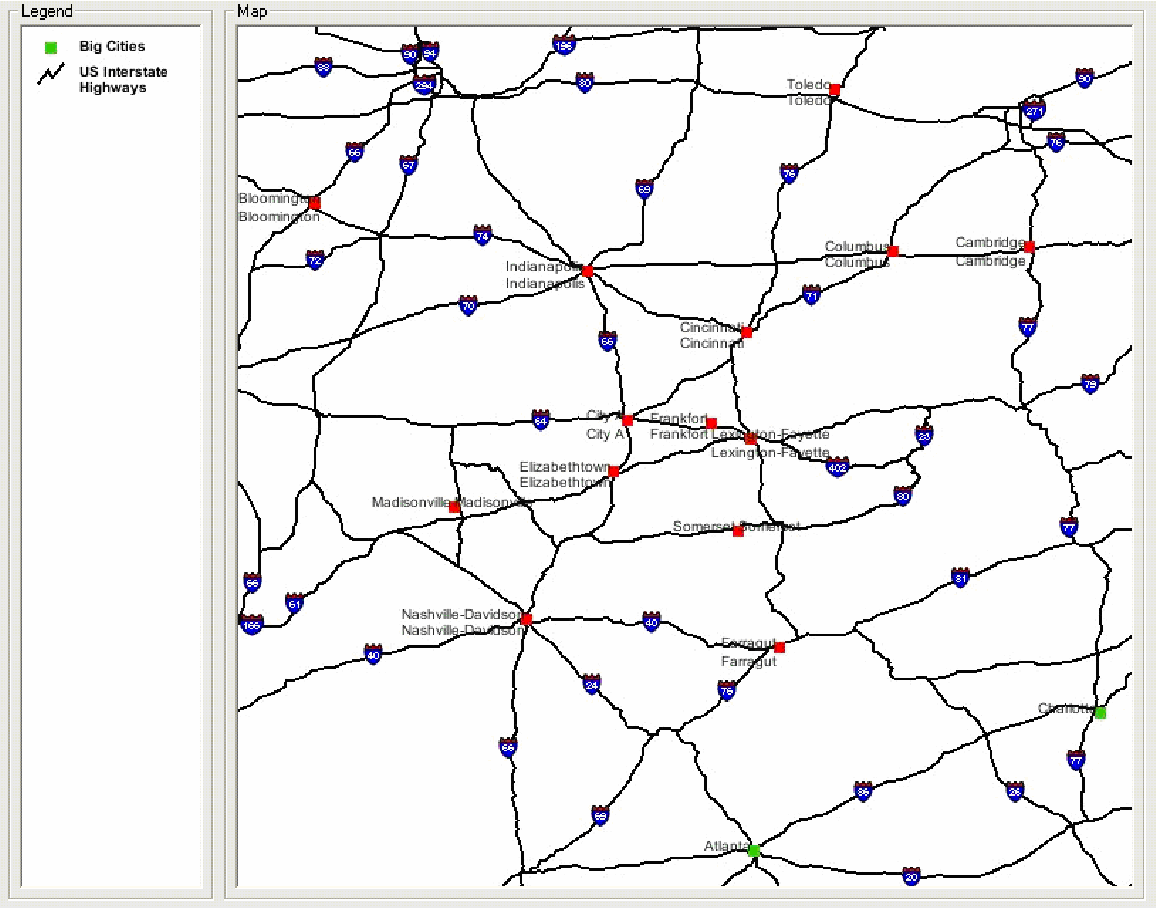


Figure **6-8**: Map Response in G (1) in **Scenario 1**

The user updates his context knowledge, in particular the spatial context knowledge, from G (1). His current context knowledge is shown in Figure **6-9** referring to the *PlanGraph* representation. Comparing the GIS agent's current context knowledge in Figure **6-7** and the user's in Figure **6-9**, we can see that the GIS agent and the user's understandings about the context behind the user's spatial information request in U(1) are different from the top level of the context structure, that is, the context-dependency problem in this scenario is in **Situation A**. Although the GIS agent's model of the meaning of *near* is similar to the user's, the GIS agent's understanding about *near* is still different from the user's.

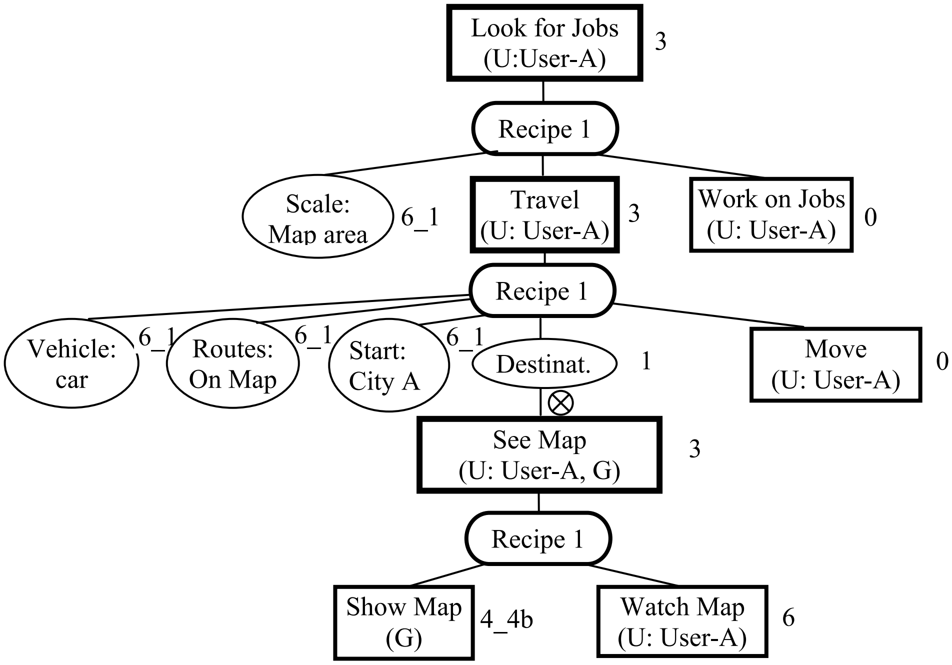
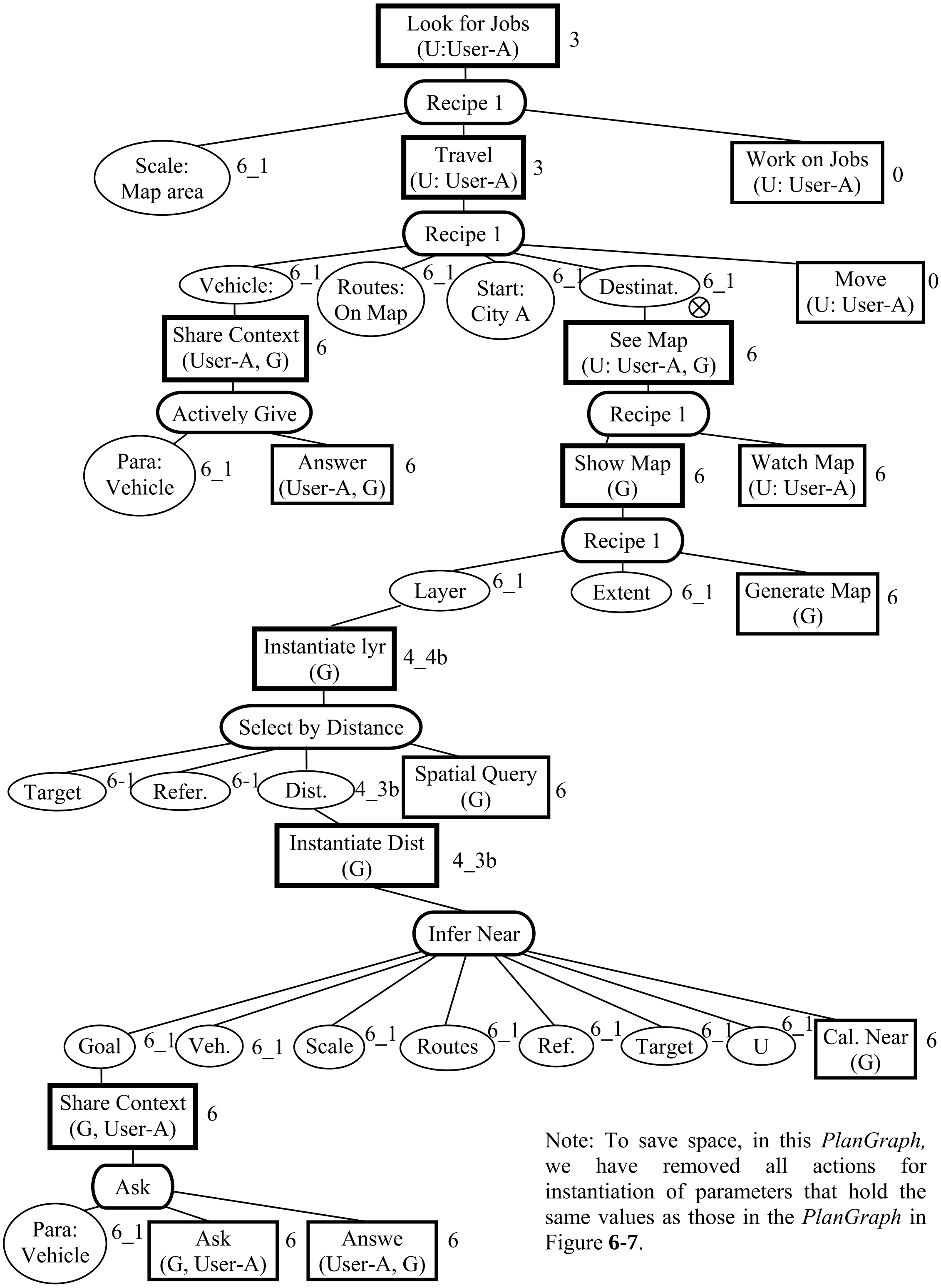


Figure **6-9**: User's context knowledge after G (1) in **Scenario 1**

Handling the Context-Dependency Problem

Upon receiving the user's negative response in U(2), the GIS agent updates its MSN on *See Map* as 4_4a because it believes that *See Map* needs to be improved. By applying **EAA4_4a** and **EAP4_4a**, which are specifically designed for vagueness handling, the GIS agent re-visits the existing *SharedPlan* in *PlanGraph*, and traces back to the parameter *Dist* in *Select by Distance*. By applying **EAP4_2(a)** and **EAA4_2(b),** the GIS agent re-visits *Dist* with the previous plan *Infer Near* to instantiate *Dist.* When trying to instantiate the parameter *Goal* under *Infer Near*, by applying **EAP1(b)**, which is specifically designed for handling the un-shared context information here, the GIS agent realizes that the user's goal behind the user's spatial information request in U (1) is not shared. Thus, the GIS agent initiates a context-sharing dialogue in G (2) with the recipe *Ask* of *Share Context* in Figure **A-11** (b) to ask the user for such information: "What is your goal in asking for this map?" and waits for the user's answer.

After receiving the user's answer in U (3) about the user's goal, *Look for Jobs*, the GIS agent re-instantiates the parameter *Goal* and elaborates and integrates the action *Look for Jobs* with the existing *PlanGraph* by attaching *See Map* with the parameter *Destination* in the sub action *Travel* under *Look for Jobs*. As to the second half of U (3) about the user's transportation mode, the GIS agent interprets it as the user's active context-sharing dialogues with the recipe *Actively Give* of *Share Context* in Figure **A-11** (c). The GIS agent attaches *Actively Give* with the parameter *Vehicle* under *Travel*. Consequently, the GIS agent re-estimates *near* as "150 miles" with *Infer Near* with new contextual information retrieved from *PlanGraph*. Up to now, all context information used in *Infer Near* cannot be improved better (to the GIS agent, the GIS agent usually does not directly communicate the spatial size information to the user). Thus, the GIS agent updates its MSN on *Infer Near* and *Dist* as 4_3b. The GIS agent re-generates a map response with the new spatial distance value and send it out in G (3): "Here are the cities within 150 miles of City A, is this OK?". The current *PlanGraph* and the map response in G (3) is shown in Figure **6-10** and Figure **6-11**.



Note: To save space, in this *PlanGraph,* we have removed all actions for instantiation of parameters that hold the same values as those in the *PlanGraph* in Figure **6-7**.

Figure **6-10**: *PlanGraph* after G (3) in **Scenario 1**

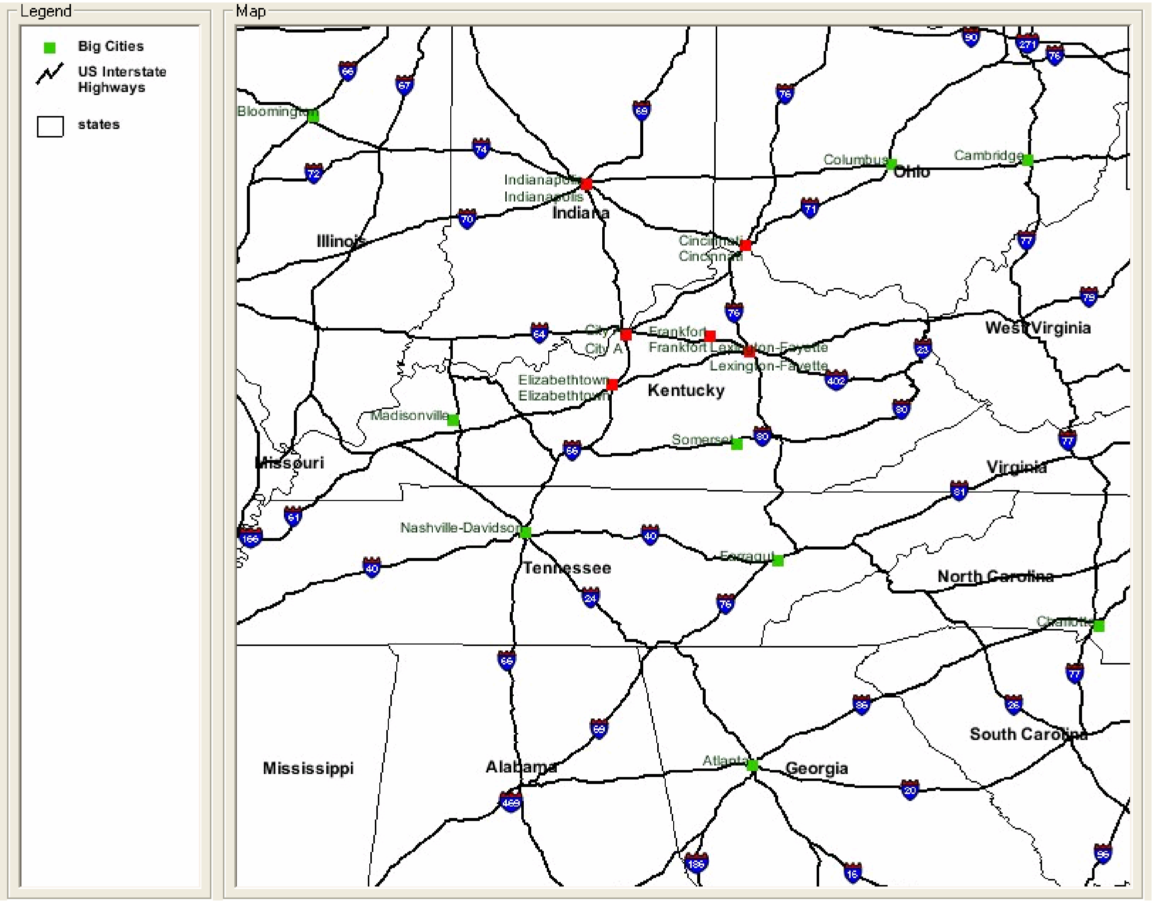


Figure **6-11**: Map response in G (3) in **Scenario 1**

By comparing the system's context knowledge after G (3) (by referring to the root plan, *Look for Jobs*, in Figure **6-10**) and the user's context knowledge after G (1) (Figure **6-9**), we can see that the system has achieved a shared understanding about the current context with the user through two different context-sharing collaborative dialogues between G (2) and U (3). One is initiated by the system to ask the user for missed contextual information, and the other is initiated by the user to tell the system certain contextual information without being asked. In this scenario, under constraints of the shared context, the system and the user reach a shared understanding about the vague spatial concept *near*. This is because in this scenario, the system and the user have similar models of the meaning of *near*.

Finally, after receiving the user's positive response in U (4), the GIS agent believes that the human-GIS collaboration on the user's *See Map* spatial intention is successfully done. So, it updates all its MSNs on *See Map* and its descendants in the *PlanGraph* in Figure **6-10** as successful.

6.3 Scenario 2: Handling the Context-Dependency Problem in Situation B

Scenario 2, which involves the context-dependency problem in **Situation B**, is shown in Figure **6-12**. Same as **Scenario 1**, in this scenario, the GIS agent's model of the meaning of the vague spatial concept includes the same list of contextual factors as those in the user's model.

G: *GeoDialogue*   U: User (User-A)

Before the following conversation, the map on the GIS shows a basic map of City C (Figure **6-13**).

U (1): Hi, I am User-A. I need to buy some foods. Show me grocery stores near Apartment A.
G (1): (highlight certain food stores near Apartment A) By driving, here are grocery stores within 3 miles of Apartment A. Is this OK to you?
U (2): No. I plan to walk.
G (2): Here are grocery stores within 1 mile to Apartment A. Is this OK?
U (3): Yes, thank you.

Figure **6-12**: **Scenario 2**

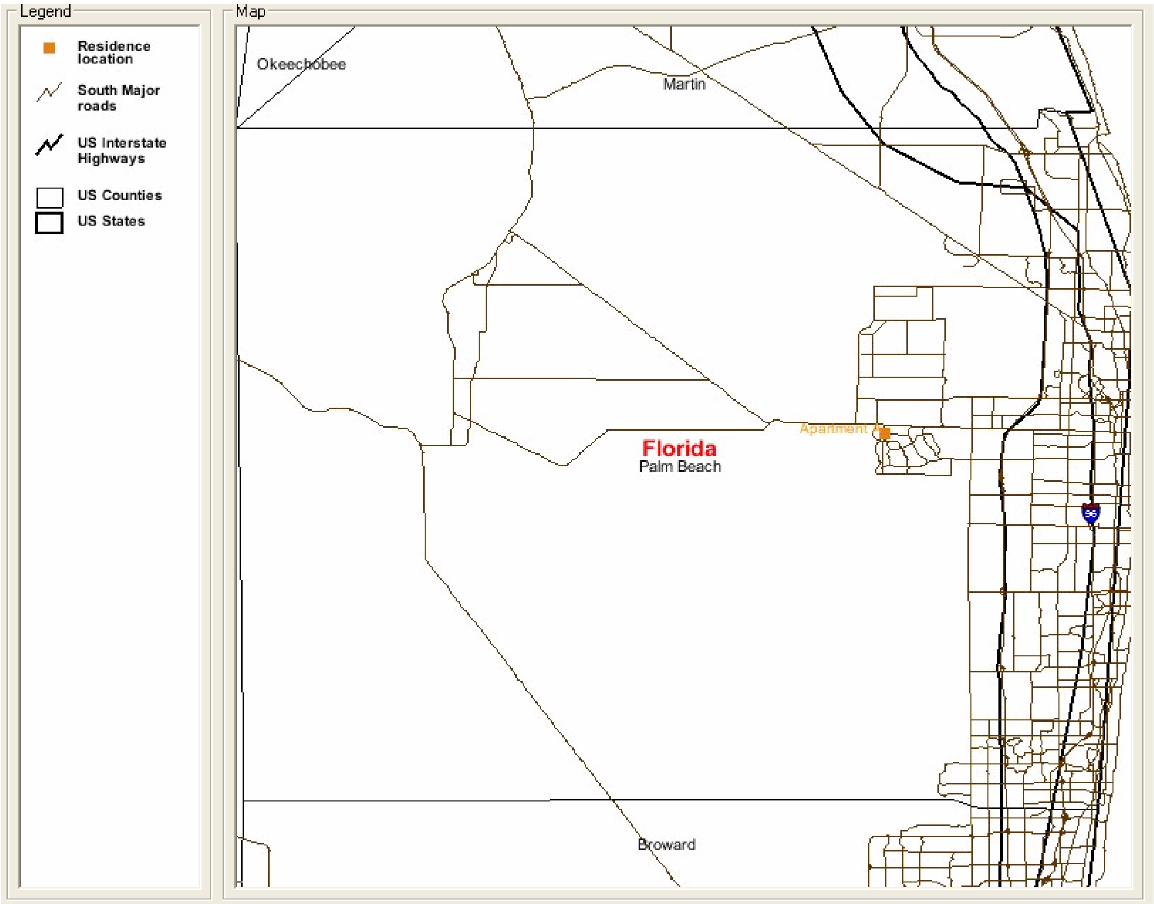


Figure **6-13**: Initial Map in **Scenario 2**

6.3.1 Extension of the *PlanGraph* Model

To enable the GIS agent to handle the context-dependency problem in **Scenario 2**, we have further extended actions and their recipes and reasoning algorithms of the *PlanGraph* model described in Chapter 4.

Extension of Actions and Recipes

To enable the GIS agent to understand the food shopping context involved in this scenario, we have designed and implemented the actions and recipes in Figure **A-3** by following the general actions and recipes in Figure **4-10**. Similarly, in this scenario, the user's *See Map* intention is attached to the parameter *Destination* in the subaction *Travel* because the grocery stores that the user is trying to identify in *See Map* are the destination locations in the user's traveling activity.

Extension of Reasoning Algorithms

By following the general rule of designing the GIS agent's reasoning algorithms described in Chapter 4, we have extended the reasoning algorithms described and implemented in Chapter 5 to facilitate the GIS agent to handle the vagueness problem in **Scenario 2**.

In this scenario, after receiving the first half of U (1) about the user's food shopping intention, the GIS agent recognizes the user's intention as *Food Shopping*. Although *Food Shopping* does not directly involve the GIS agent's participation, it does involve parameters representing spatial context information, which the user may ask the system to identify. So, this action is taken as the context involving the user's spatial information request to the GIS. To enable the GIS agent to interpret such actions as contexts of human-GIS collaboration, we have designed and implemented **IA2** (see Appendix C) for the GIS to interpret such actions and initiate the *PlanGraph* with such actions.

The user's spatial intention *See Map* underlying the second half of U (1) about the user's spatial information request may not directly relate to the current attention focus in

PlanGraph, *Food Shopping*, but can relate to it after certain steps of elaboration about *Food Shopping*. To enable the GIS agent to interpret such actions, we have designed and implemented **IA6** (see Appendix C).

In this scenario, the user's active context-sharing dialogue in U (2) about the transportation mode can be interpreted as the action *Share Context*. This action does not directly relate to the current focus in *PlanGraph*, *See Map*, but relates instead to an ascendant of the current focus. We have also designed and implemented **IA7** (see Appendix C) for the GIS agent to reason about how to interpret such actions and integrate them with the existing *PlanGraph*.

6.3.2 Process of Handling the Context-Dependency Problem in Situation B

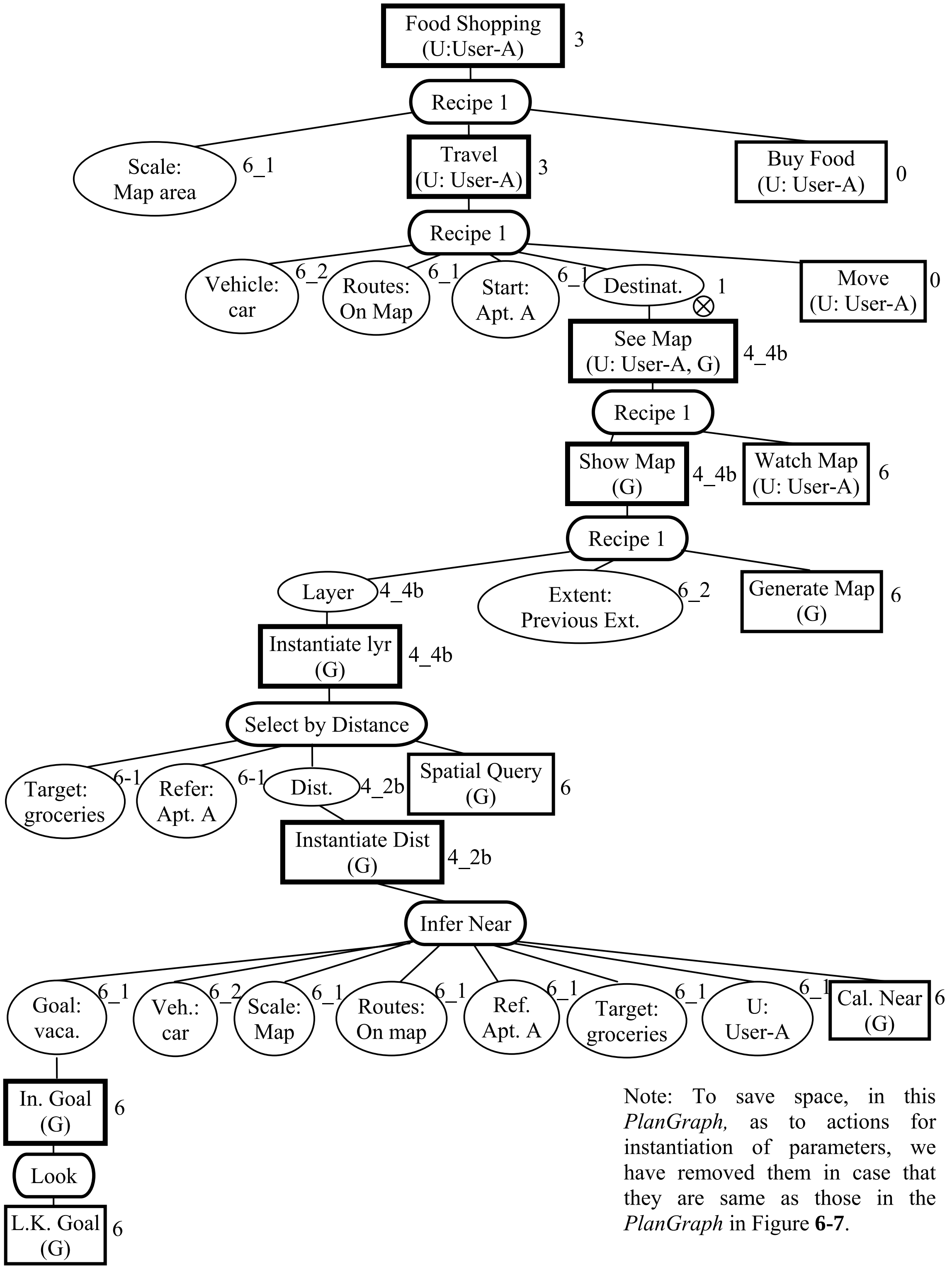
Initial Shared Context before Communication

In this scenario, before U (1), the shared context between the GIS agent and the user agent is same as that in **Scenario 1**.

Building Shared Context through U(1) and G (1)

After receiving the first half in U (1) about the user's intention about food shopping, by applying **IA2**, which is specifically designed for interpreting the user's intention about the context, the GIS agent initiates *PlanGraph* with *Food Shopping* after elaborating it with the recipe in Figure **A-3** (a) and then focuses attention on it. After receiving the second half in U (1) about the user's spatial information request, the GIS agent interprets it as the user's spatial intention *See Map*. By applying **IA6**, which is specifically designed for integrating the user's spatial intention with the plan referring to the context represented in *PlanGraph*, the GIS agent elaborates *Food Shopping* with available information (without inferring any contextual information) and attaches it to the parameter *Destination* in *Travel* under *Food Shopping*.

With a reasoning process similar to that in **Scenario 1** for the GIS to elaborate *See Map*, *GeoDialogue* recursively elaborates it until the GIS agent tries to instantiate the optional parameters in the plan *Infer Near* representing contextual factors. Similarly, the GIS agent tries to elaborate the parameters or actions in the action *Food Shopping* corresponding to these parameters in *Infer Near* first. In the context of *Food Shopping*, the GIS agent infers the spatial size of the area as the existing map extent area and the transportation mode is driving. After elaborating *Food Shopping*, the GIS agent estimates *near* as three miles based on the GIS agent's knowledge of User-A's understanding about *near* in the food shopping context, generates a map response in G (1), and sends it out to the user in G (1). The current *PlanGraph* after G (1) and the map response in G (1) are shown in Figure **6-14** and Figure **6-15**, respectively.



Note: To save space, in this *PlanGraph,* as to actions for instantiation of parameters, we have removed them in case that they are same as those in the *PlanGraph* in Figure **6-7**.

Figure **6-14**: *PlanGraph* after G (1) in **Scenario 2**

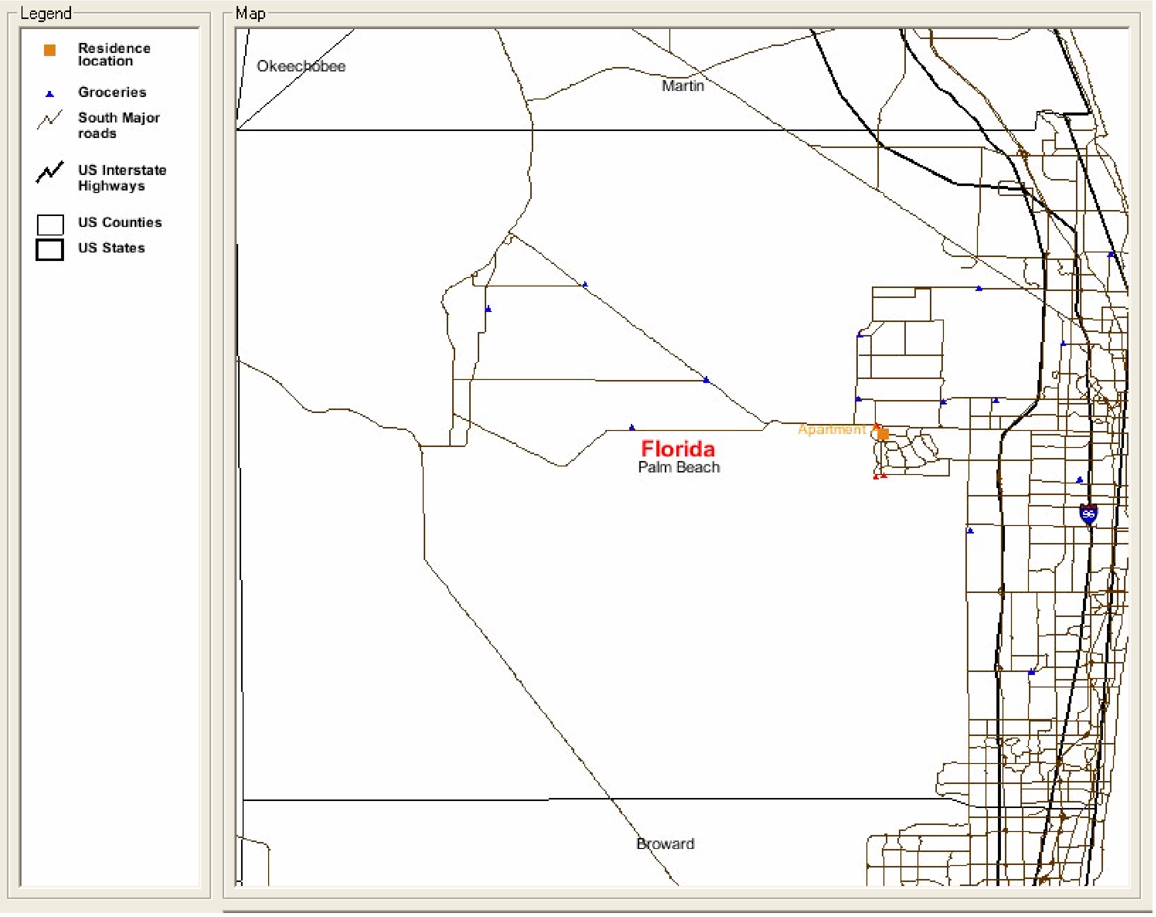


Figure **6-15**: Map Response in G (1) in **Scenario 2**

The user's current context knowledge after receiving G (1) is shown in Figure **6-16**. Comparing the GIS agent's current context knowledge in Figure **6-14** and the user's in Figure **6-16**, we can see that the GIS agent and the user's understandings about the context behind the user's spatial information request in U(1) are different at only instantiation of the transportation mode involved in the context structure, that is, the context-dependency problem in this scenario is in **Situation B**. Similar to **Scenario 1**, in this scenario, although the GIS agent's model of the meaning of *near* is similar to the user's, the GIS agent's understanding about *near* is still different from the user's.

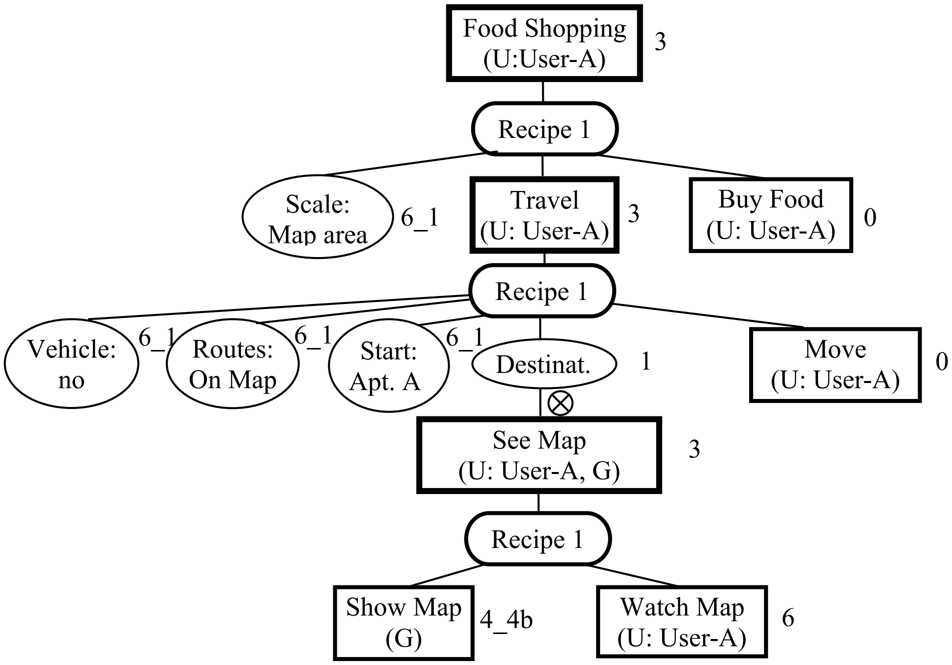
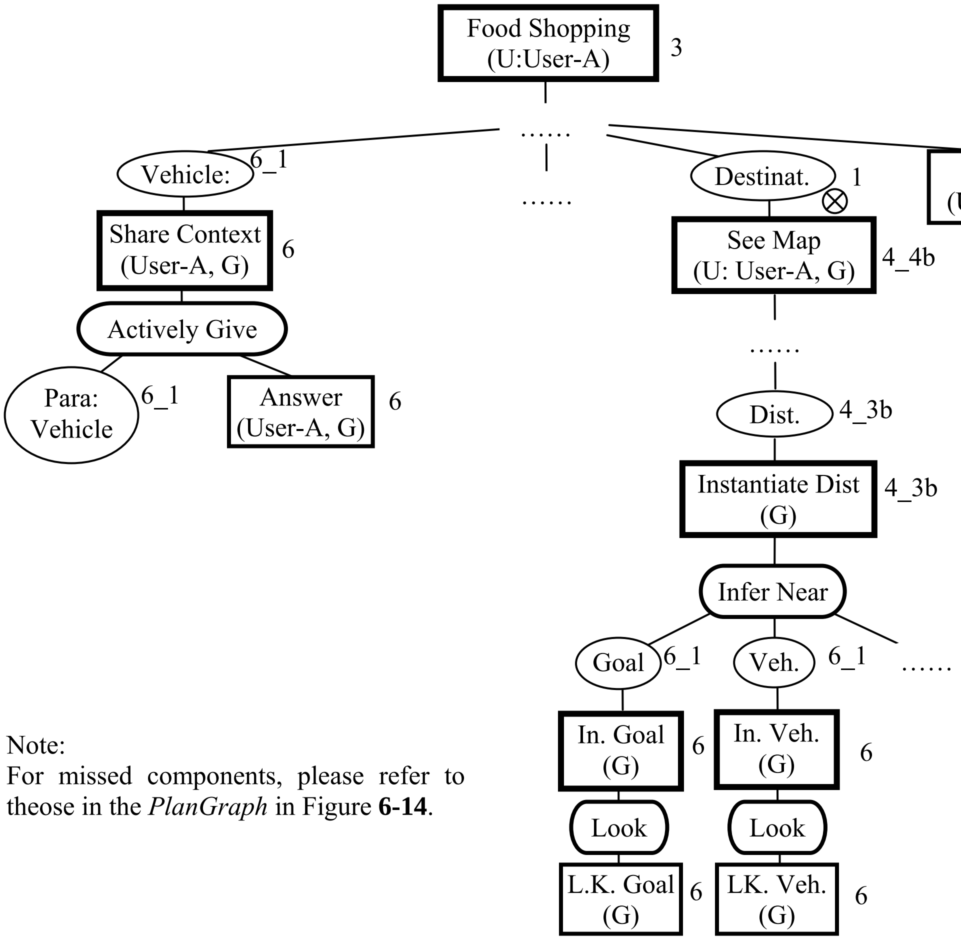


Figure **6-16**: User's Context Knowledge after G (1) in **Scenario 2**

Handling the Context-Dependency Problem

Upon receiving the user's negative response "No" in U(2), the GIS agent updates its MSN on *See Map* as 4_4a because it believes that *See Map* needs to be improved. As to "I plan to walk" in U (2), the GIS agent interprets it as the user's active context-sharing dialogue, represented as the recipe of *Actively Give* of *Share Context*. By applying **IA7**, which is specifically designed for the GIS to understand the user's actively context-sharing dialogues, the GIS agent integrates *Actively Give* with the parameter *Vehicle* in *Travel* under *Food Shopping*. After updating the parameter value of *Vehicle*, the GIS agent moves attention back on elaborating *See Map* with MSN 4_4a.

By using the same reasoning process as that in **Scenario 1** for the GIS to elaborate *See Map* with MSN 4_4a, the GIS agent traces back to improve the meaning of *near* with the previous plan *Infer Near*. After successfully retrieving the updated transportation mode information from the current *PlanGraph*, the GIS agent re-infers *near* as one mile, re-generates a new map response and sends it out in G (2): “Here are the grocery stores within 1 mile of Apartment A. Is this OK?”. The current *PlanGraph* and the map response in G (2) are shown in Figure **6-17** and Figure **6-18** .



Note:
For missed components, please refer to theose in the *PlanGraph* in Figure **6-14**.

Figure **6-17**: *PlanGraph* after G (2) in **Scenario 2**

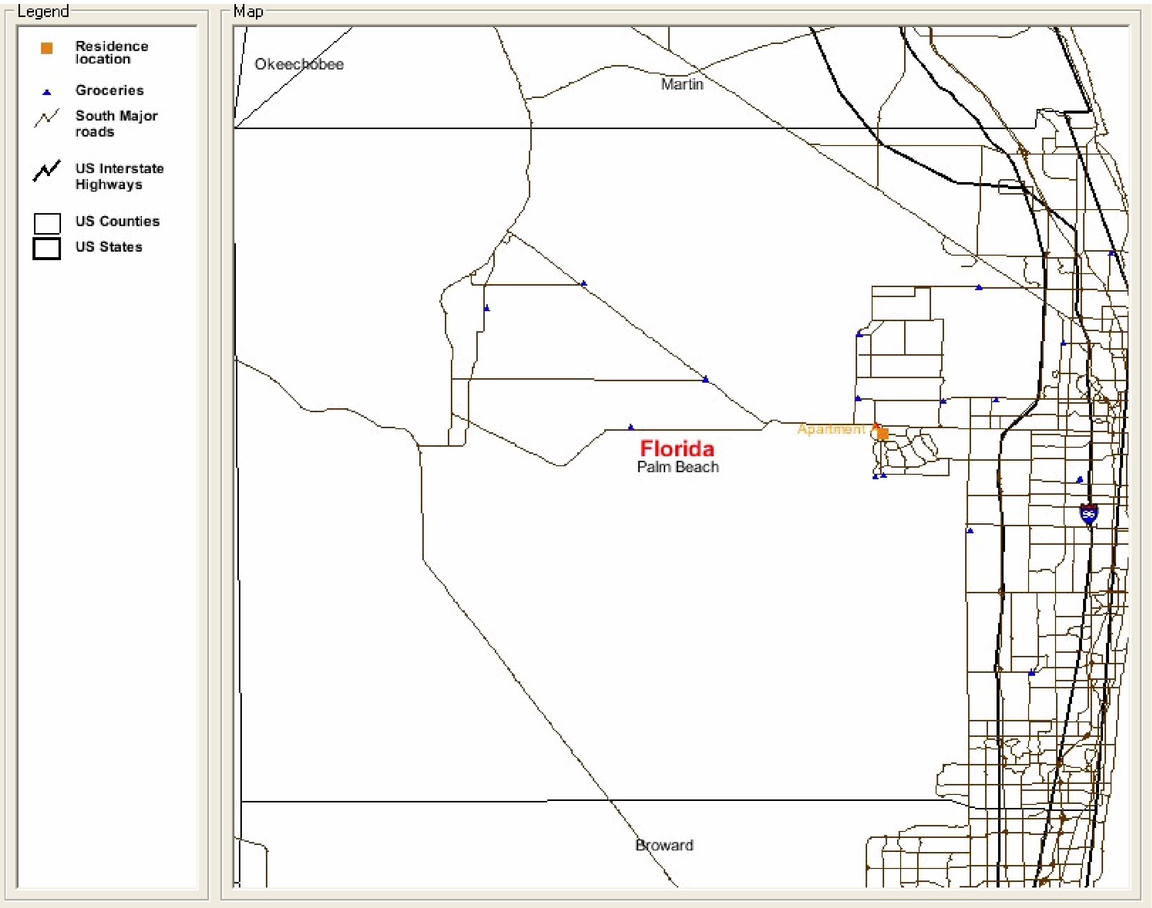


Figure **6-18**: Map Response in G (2) in **Scenario 2**

By comparing the system's context knowledge after G (2) (with referring to the root plan, *Food Shopping*, in Figure **6-17**) and the user's context knowledge after G (1) (Figure **6-16**), we can see that the system has achieved a shared understanding about the current context with the user through the context-sharing collaborative dialogue in U (3). U (3) is initiated by the user to actively share the task contextual information with the system without being asked. Similar to **Scenario 1**, under constraints of the shared understanding about the current context, the system finally reaches a shared understanding about the vague spatial concept *near* with the user because the two agents have similar models of the meaning of *near*.

Finally, after receiving the user's positive response in U (3), the GIS agent updates all its MSNs on *See Map* and its descendants in the *PlanGraph* in Figure **6-17** as successful.

6.4 Scenario 3: Handling the Context-Dependency Problem in Situation C

Scenario 3, which involves the vagueness problem in **Situation C**, is shown in Figure **6-19**. In this scenario, the GIS agent's model of the meaning of the vague spatial concept *near* includes contextual factors different from those in the user's model.

G: GIS U: User (User-A)

Before the following conversation, the map on GIS shows a basic map of City A (including state boundaries, see Figure **6-20**).

U (1): Hi, I am User-A. I am planning to visit cities near *City A* by driving on weekends. Show me cities *near* City A.
G (1): (show interstates, City A and all cities in the current area, and highlight certain cities *near* City A) Here are cities within 300 miles of City A. Is this OK?
U (2): (Pointing to an area on the map) No, please zoom to this area first.
G (2): (Show a new map) Here is the map after zooming. Is this OK?
U (3): No. Please show me history museums and outlets in these cities.
G (3): Here you are. Is this OK?
U (4): (Point to the map with gesture) No. Please highlight these cities only.
G (4): Here are the cities highlighted.
U (5): Thank you.

Figure **6-19**: **Scenario 3**

Figure **6-20**: Initial Map in **Scenario 3**

6.4.1 Extension of the *PlanGraph* Model

By following the general rule about design of reasoning algorithms in the *PlanGraph* model, we have further extended this model for the GIS agent to handle the vagueness problem in **Scenario 3**.

In this scenario, the user sends a new spatial information request in U (3), which is different from the current spatial information request on focus. It does not directly relate to the current elaboration of the context behind the current focus represented in *PlanGraph*. To enable the GIS agent to interpret such intentions, we have designed and implemented **IA9** for the GIS agent to take the user's new spatial information request as initiating a context-sharing dialogue and shifts attention back to the previous spatial intention after returning such information to the user.

In this scenario, the initial map does not include spatial information about the routes needed in the user's planning for the vacation purpose and for the GIS agent's understanding of *near*, and neither does the user request it in U (1). Thus, we have designed and implemented part of **EAP1(a)** for the GIS agent to actively share such spatial information with the user in order to build a shared context with the user.

6.4.2 Process of Handling the Context-Dependency Problem in Situation C

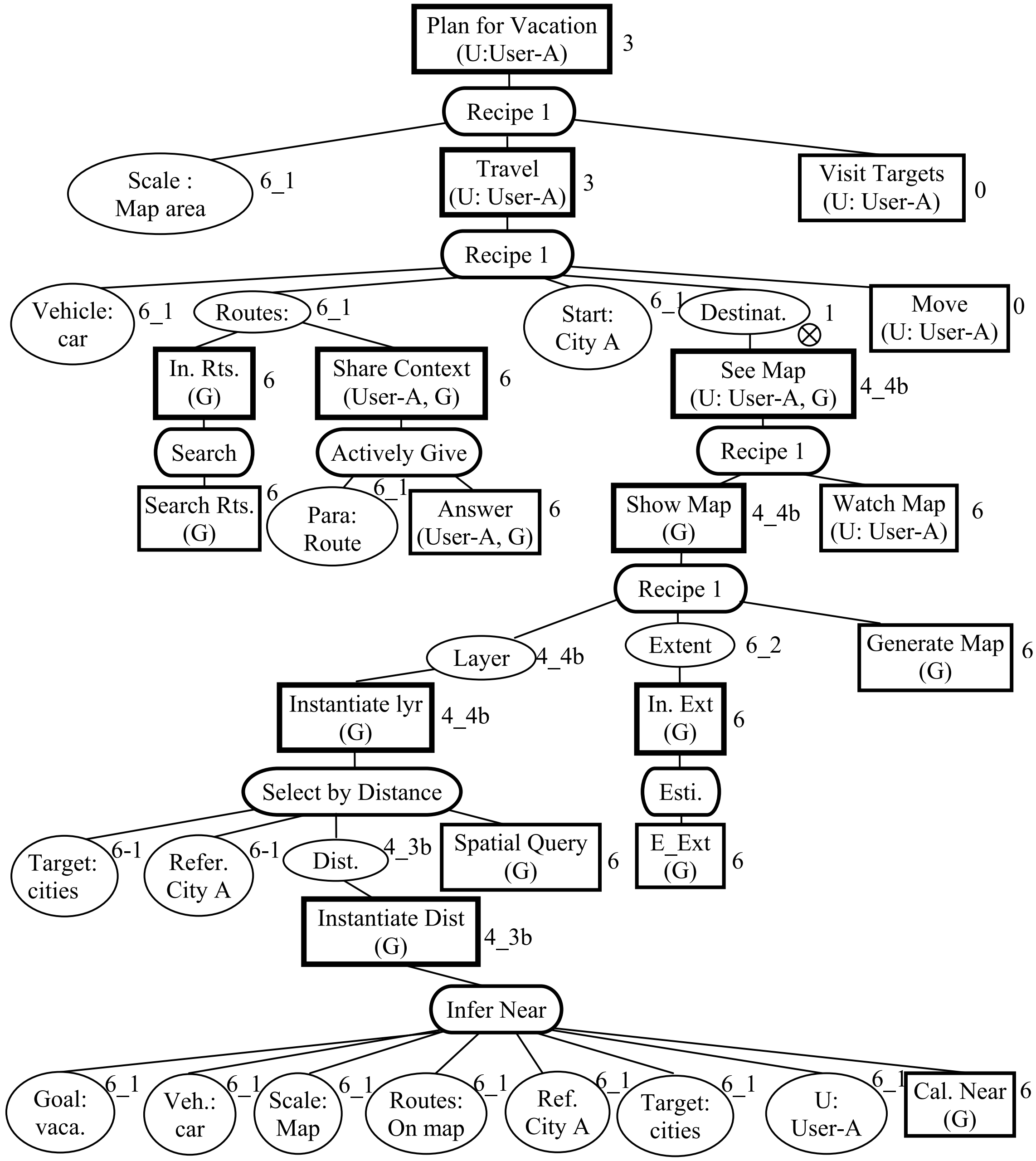
Initial Shared Context before Communication

In this scenario, before U (1), what the GIS agent and the user agent shared about the context to be involved in communication about the vague spatial concept *Near* is similar to those in **Scenario 1** and **Scenario 2**. In this scenario, the initial map contains less spatial context information than those in the previous two scenarios.

Building Shared Context through U(1) and G (1)

After receiving U (1), the GIS agent initiates *PlanGraph* with *Plan for Vacation* expressed in the first half of U (1), further elaborates it so that the user's intention *See Map* expressed in the second half of U (1) can be integrated with it.

With a reasoning process similar to that for the GIS to elaborate *See Map* after receiving U (1) in **Scenario 2**, *GeoDialogue* recursively elaborates it, and generates a map response and speech response in G (1). One difference between this scenario and the **Scenario 2** lies in the process of elaboration of the action representing the context before elaboration of the parameter *Routes* in *Infer Near*. During the process of elaboration *Plan for Vacation* in this scenario, by applying **EAP1(a)** specifically for the GIS agent to actively share context information that the user may not know with the user, the GIS agent attaches the recipe *Actively Give* of *Share Context* with the parameter *Routes* in the action *Travel* involved in *Plan for Vacation* and executes it after successfully instantiating this parameter. Another difference is on MSNs on *Dist* and *Instantiate Dist*. In this scenario, MSNs on these two nodes are 4_3b because all context information used in *Infer Near* can not be further improved, while they are 4_2b in **Scenario 2**. With the context information retrieved from *Plan for Vacation*, the GIS agent generates a map showing cities within 300 miles of City A. The current *PlanGraph* after G (1) and the map response in G (1) are shown in Figure **6-21** and Figure **6-22**, respectively.



Note: To save spaces, we have removed part of actions for instantiation of parameters with MSN 6_1

Figure **6-21**: *PlanGraph* after G (1) in **Scenario 3**

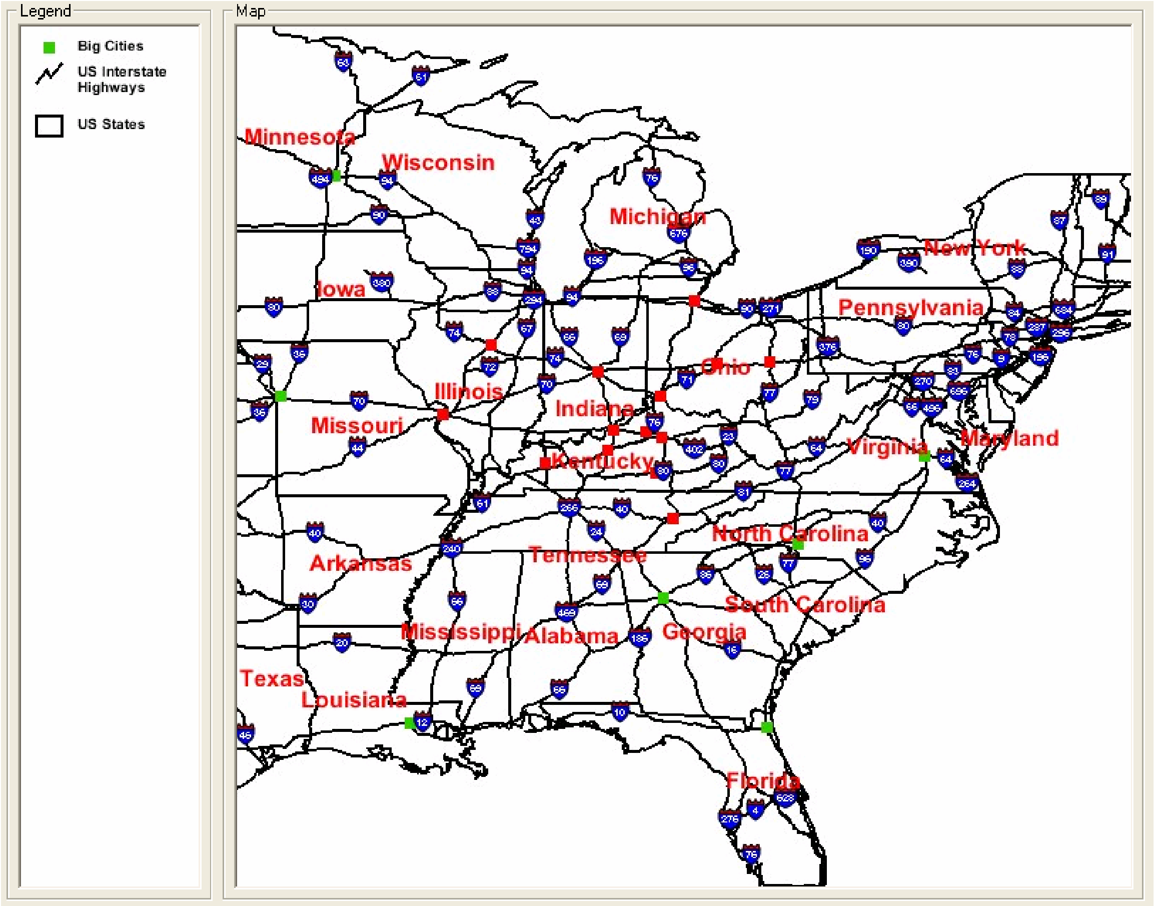


Figure **6-22**: Map Response in G (1) in **Scenario 3**

In this scenario, the user's model of the meaning of *near* involves more context factors, in addition to those in the GIS agent's model (Eq. **6.1**). The user considers influence on *near* from more details about the visiting activity, including the visiting targets, as well as the action "Visit". The user's current context knowledge after receiving G (1) is shown in Figure **6-23** by referring to the *PlanGraph* representation. Comparing the GIS agent's current context knowledge in Figure **6-21** with the user's in Figure **6-23**, we can see that the GIS agent and the user's understandings about the context are different at the depth of the visiting activity in the context structure. Thus, the context-dependency problem in this scenario is in **Situation C**.

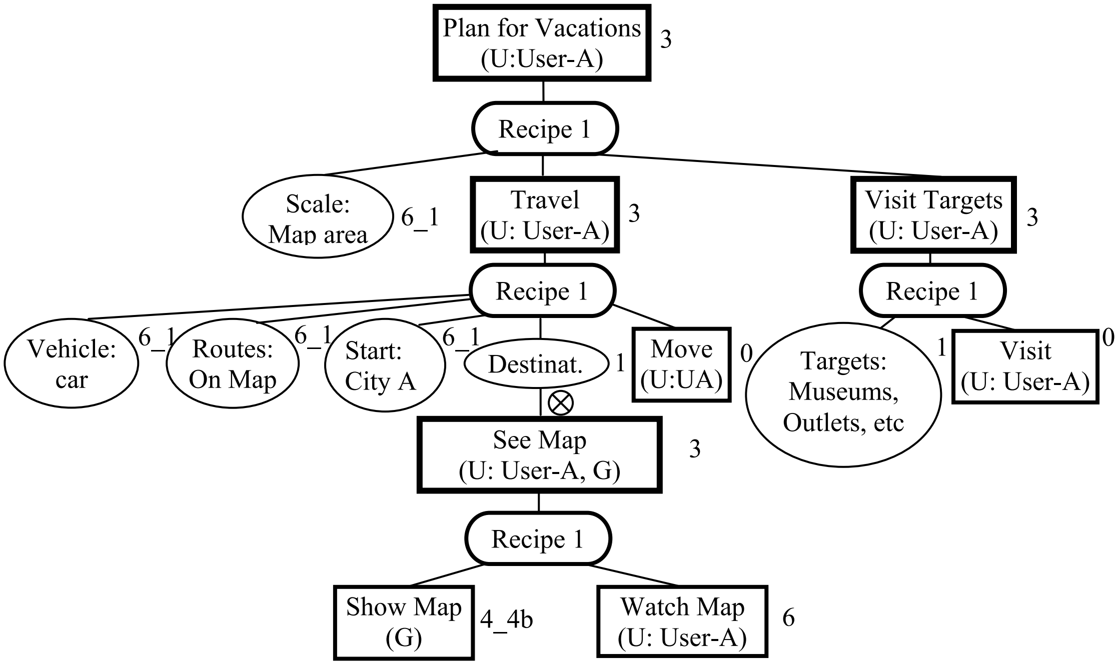
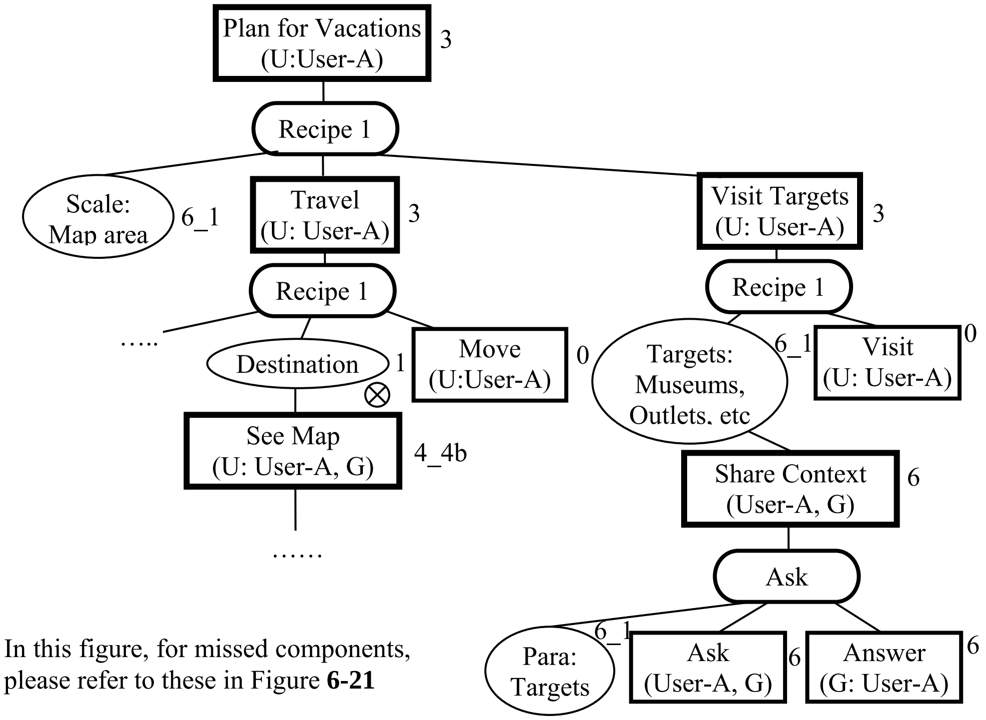


Figure **6-23**: User's Context Knowledge after G (1) in **Scenario 3**

Handling the Context-Dependency Problem

After receiving the user's negative response and the request to modify the current map extent in U (2), by applying **IA7(b)**, the GIS agent shifts attention back to the previous focus, *See Map*, involving vagueness, after successfully returning a new map response in G (2) with the new map extent required by the user.

From the new map response in G (2), the user can not find his visiting targets of history museums and outlets. So he directly asks the GIS for such information in U (3): "No. Please show me history museums and outlets in cities". Upon receiving the user's new spatial information request in U (3), by applying **IA9(b)**, which is specifically designed for the GIS to interpret the user's new spatial information request under the context of the current communication, the GIS agent re-elaborates the current plan for *Plan for Vacation*, in particular, its sub action, *Visit Targets*. From the GIS agent's knowledge base, the GIS agent knows that history museums and outlets are part of visiting targets in the plan for *Visit Targets*. So the GIS agent further elaborates *Visit Targets*. By applying **IA9(b)**, the GIS agent takes the user's spatial information request in U (3) as initating a context-sharing dialogue with the recipe, *Ask*, of *Share Context*. After successfully showing such spatial context-information to the user in G (3) to the user, the GIS agent comes back to the previous attention focus, *See Map*, for "cities near City A", involving vagueness. The current *PlanGraph* showing the GIS agent's current context knowledge is shown in Figure **6-24**.



In this figure, for missed components, please refer to these in Figure **6-21**

Figure **6-24**: *PlanGraph* after G (3)

After receiving the map response in G (3), the user updates his spatial context knowledge on the visiting targets, that is, the parameter node *Targets* in Figure **6-23**. Comparing the system's current context knowledge in Figure **6-24** and the user's current context knowledge, we can see that they have reached a shared understanding about the context behind the user's spatial information request in U (1).

Further Reducing Vagueness after Building the Shared Context

Although, after G (3), the user and the system have a shared understanding about the current context, the user does not agree with the system's understanding about the concept *near*. The user points out the destination cities with a gesture in U (4) based on his current understanding about *near* with updated spatial contextual information represented in the map in G (3). In the current implementation of *GeoDialogue*, the GIS agent does not have knowledge about how to understand *near* based on all context information considered by the user in this scenario. Thus, the system can not generate a new map to the user.

The user's spatial intention in U (4) is interpreted by the system as another way to improve the current focus, *See Map* with MSN 4_4b. Thus, the system updates its MSN on *See Map* as 4_4a and re-visits *See Map* with another plan to instantiate the parameter *Layer* under *Show Map*. The new plan is *Select by Gesture* of *Instantiate Lyr* (Figure **A-8** (j)). The new plan does not involve vagueness, so, the GIS agent updates all its MSN on *Layer* and ascendants of *Layer* as successful after re-generating a new map response in G (4). The final map response in G (4) is shown in Figure **6-25**.

The user's positive response in U (5) confirms the GIS agent's work in G (4). In this scenario, what is different from the first two scenarios is that the two agents do not reach a shared understanding about the vague spatial concept *near* when they have reached a shared understanding about the current context. In this scenario, to further reduce vagueness, the user directly tells the system what he thinks are *near* through a gesture in U (4). Finally, the two agents successfully reach a shared understanding about the vague spatial concept after further reducing vagueness through the direct communication strategy.

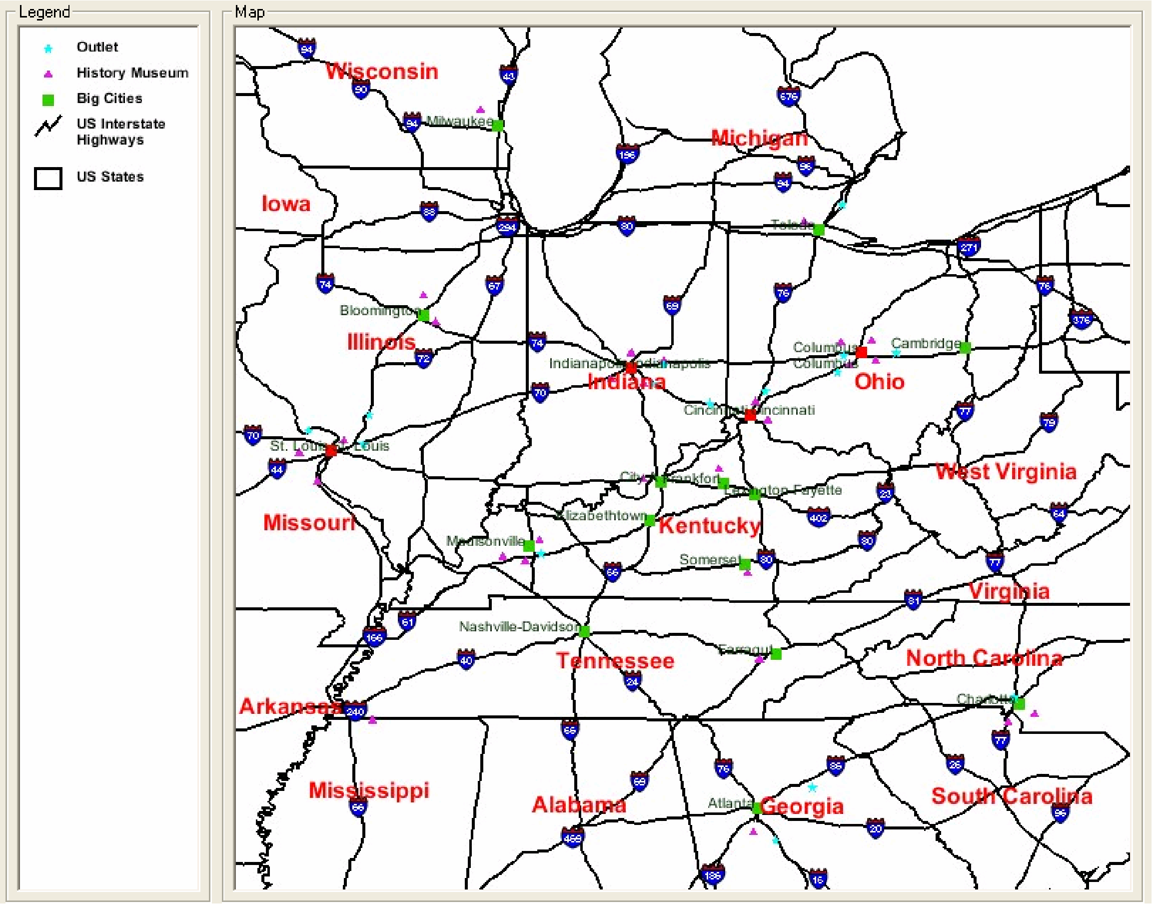


Figure **6-25**: Map Response in G (4) in **Scenario 3**

6.5 Scenario 4: Handling the Context-Dependency Problem in Situation D

Scenario 4, which involves the vagueness problem in **Situation D**, is shown in Figure **6-26**. In this scenario, the user's model of the meaning of the vague spatial concept *near* involves more contextual factors than those in the GIS agent's model.

G: GIS U: User (User-B)

Before the following conversation, the map on the GIS shows a basic map of City C, including state boundaries, county boundaries, interstates and major roads (see Figure **6-27**).

U (1): Hi, I am User-B. I am planning to buy food by driving. Show me grocery stores near Apartment A.
G (1): (map) Here are the grocery stores within 5 miles of Apartment A. Is this OK for you?
U (2): No.
G (2): Do you want to increase or decrease the spatial distance of *near*?
U (3): Decrease, please.
G (3): (map) Here are the grocery stores within 2.5 miles of Apartment A. Is this OK?
U (4): No. Please highlight all these grocery stores.
G (4): (map) Here are the grocery stores highlighted as you requested.
U (5): Thanks.

Figure **6-26**: **Scenario 4**

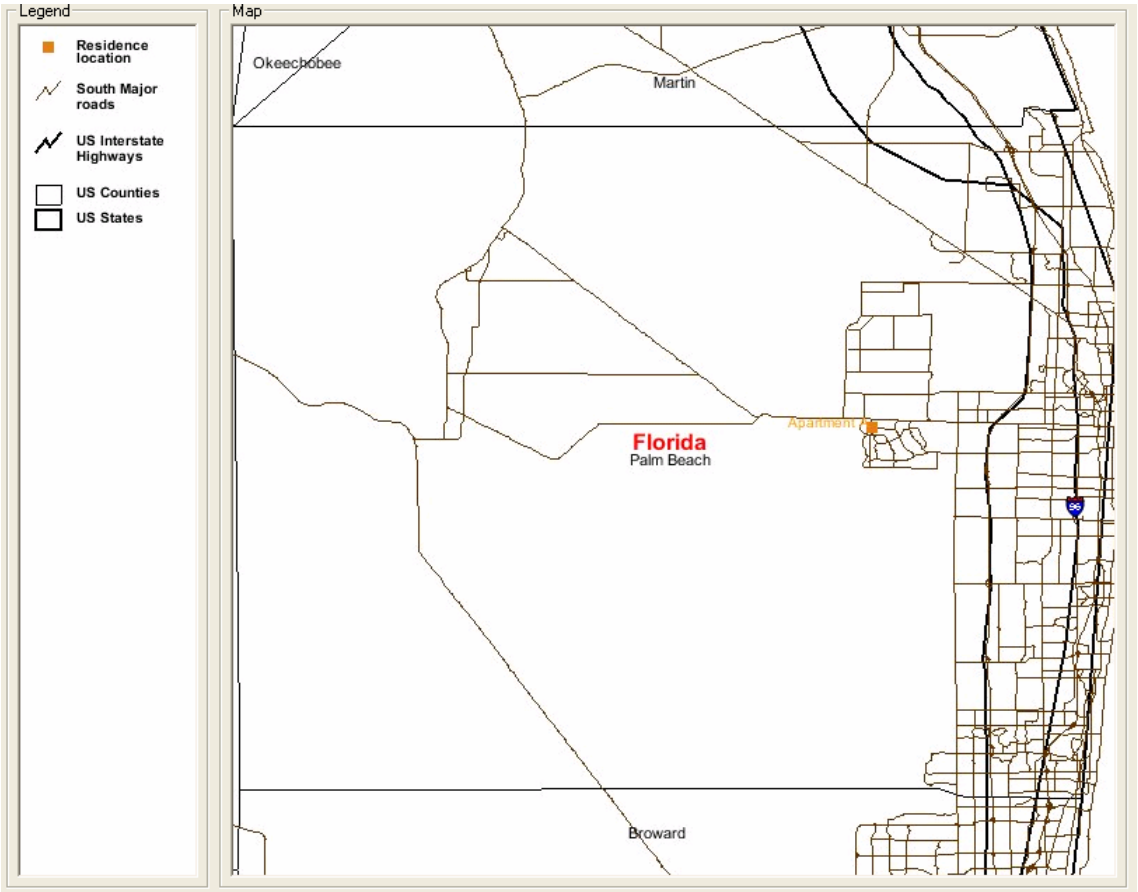


Figure **6-27**: Initial Map in **Scenario 4**

6.5.1 Extension of the *PlanGraph* Model

Similarly, we have also further extended the *PlanGraph* model represented in Chapter 4 for the GIS agent to handle the context-dependency problem in **Scenario 4**.

Extension of Actions and Recipes

To enable the GIS agent to have negotiation collaborative dialogues like the one in G (2) and U (3), we have designed and implemented the recipe *Ask* of the action

Negotiate (Figure **A-12** (b)). In order to enable the GIS agent to understand the user's active refining comments upon being asked, we have also implemented the recipe *Active Give* of *Negotiate* (Figure **A-12** (c)).

Extension of Mental States

In this scenario, after receiving the user's negative response on its understanding about *near* in G (1), the GIS agent believes that the previous plan *Infer Near* used to understand *near* and instantiate the parameter *Dist* can not be further improved. Thus, it needs to change another plan to understand *near*. To model the GIS agent's mental beliefs involved in this process, we have further exteneded and implemented MSNs in Table **5-3** and Table **5-4**. Table **6-2** and Table **6-3** show the new MSNs and their meanings.

Table **6-2**: Extended MSNs on Action α and Their Meanings

Label		Intention on doing α	Commitment on success of α	Belief (the agent being modeled believes that)
4	4_1	individual, group	Individual and mutual	The current plan for α does not succeed, and α can be re-executed with another plan.

Table **6-3**: Extended MSNs on Parameter *p* and Their Meanings

Label		Intention on instantiating *p*	Commitment on success of instantiating *p*	Belief (the agent being modeled believes that)
4	4_1	individual, group	Individual and mutual	The current plan for instantiation of p does not succeed, and p can be re-instantiated with another plan.

Extension of Reasoning Algorithms

With the new MSNs implemented above, we have also designed and extended **EAA4_1** and **EAP4_1** (see Appendix C) for the GIS agent to change another appropriate plan to re-execute an action or re-instantiate a parameter with the new MSN.

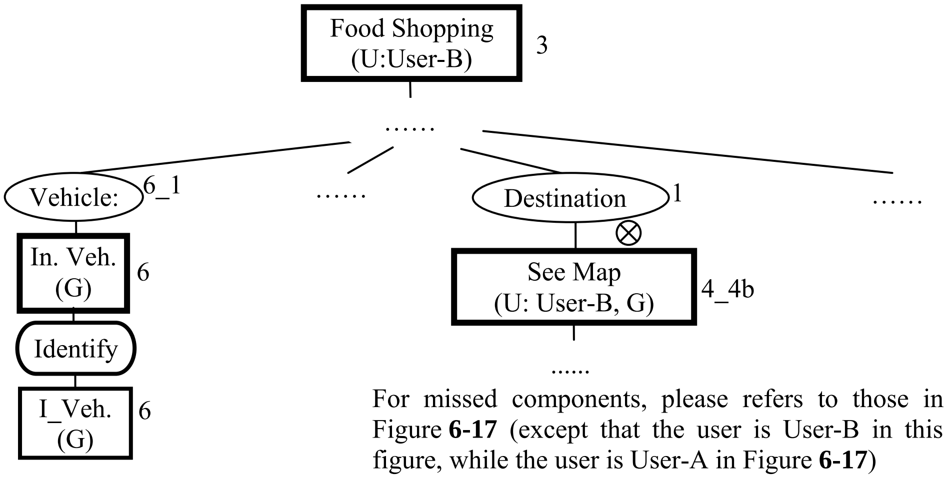
6.5.2 Process of Handling the Context-Dependency Problem in Situation D

Initial Shared Context before Communication

In this scenario, before U (1), what the GIS agent and the user agent shared about the context to be involved in communication about the vague spatial concept *Near* is similar to those in **Scenario 1** and **2**.

Building Shared Context through U(1) and G (1)

With a same reasoning process as that in **Scenario 3** for the GIS to interpret U (1), the GIS agent initiates *PlanGraph* with the plan *Food Shopping* and further elaborates it so that the user's intention *See Map* expressed in the second half of U (1) can be integrated with it. Furthermore, with a reasoning process similar to that in **Scenario 2** for the GIS to elaborate *See Map* toward its success, the GIS agent generates a map response in G (1) indicating its understanding about *near* as 5 miles. Because the user actively shares his transporation mode information in this scenario, the GIS agent updates its MSNs on *Dist* and *Instantiate Dist* as 4_3b because all context information used in *Infer Near* can not be further improved. The current *PlanGraph* is shown in Figure **6-28**. The map response in G (1) is shown in Figure **6-29**.



For missed components, please refers to those in Figure **6-17** (except that the user is User-B in this figure, while the user is User-A in Figure **6-17**)

Figure **6-28**: *PlanGraph* after G (1) in **Scenario 4**

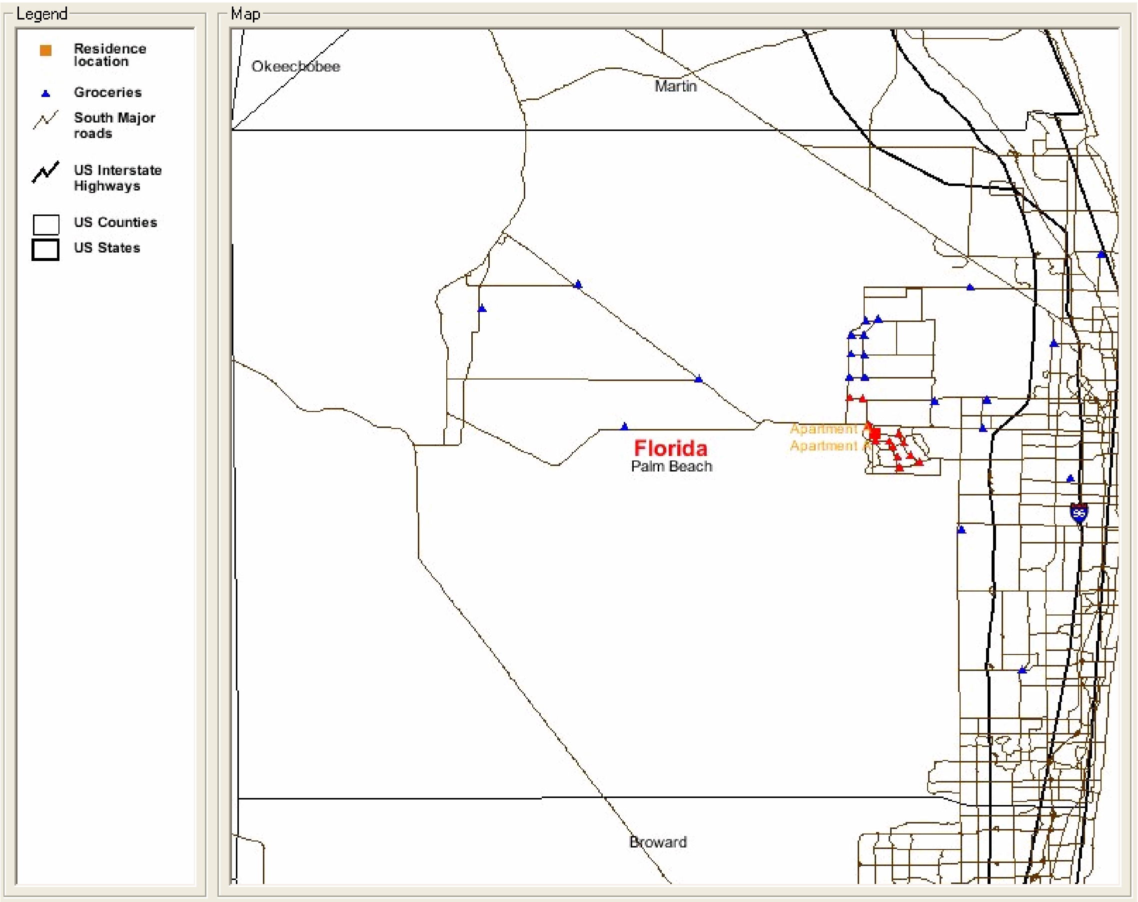


Figure **6-29**: Map Response in G (1) in **Scenario 4**

The user's current context knowledge is shown in Figure **6-30**. Comparing the current context knowledge kept by the user and the GIS agent, we can see that the two agents have different understandings about the width of *Food Shopping* and *Travel*. The user also considers influence on the meaning of *near* from the food shopping frequency, the traffic status on routes and relative spatial distances of destinations with respect to all possible destinations (*A. Pss. Dest.*) in addition to the contextual factors in the GIS agent's model of the meaning of *near*. Thus, the context-dependency problem in the current situation is in **Situation D**.

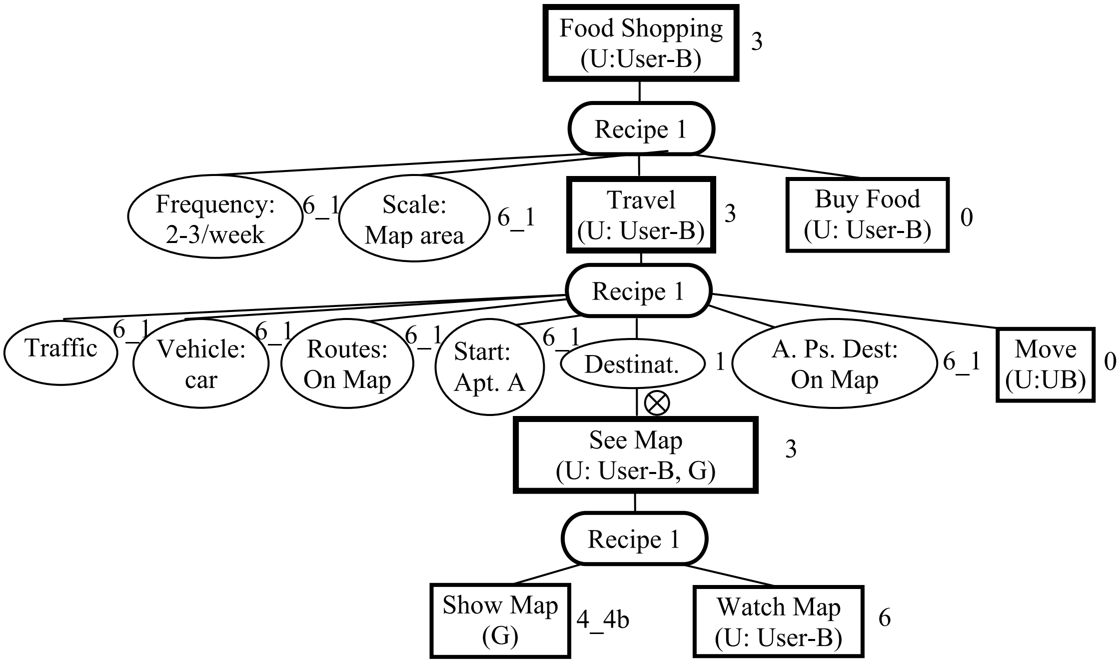


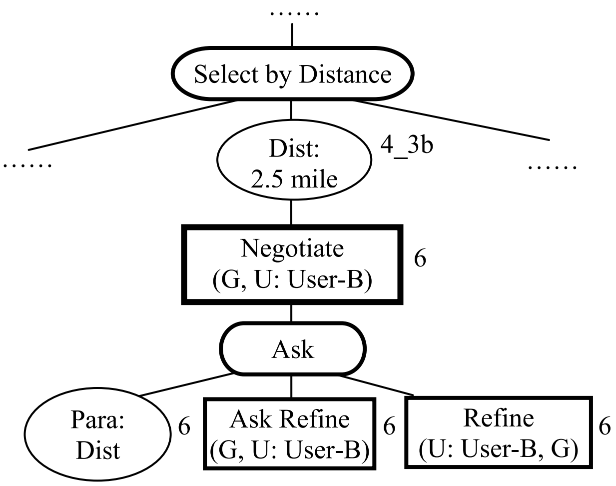
Figure **6-30**: User's Context Knowledge after G (1) in **Scenario 4**

Handling the Context-Dependency Problem

After receiving the user's negative response in U (2), the GIS agent re-visits *See Map* and traces back to the parameter *Dist* with MSN 4_3a in the plan *Select by Distance*.

At this moment, although the context-dependency problem is in **Situation D**, the system does not know this situation from the human-GIS communication. From its context knowledge kept in Figure **6-29**, the system still thinks that it has shared all contextual information used to understand the meaning of the vague spatial concept *near* with the user. By applying **EAP4_3a** and **EAP4_1**, which are specifically designed for the GIS to improve the parameter value of a parameter which previous instantiation plan fails, the GIS agent updates its MSN on *Dist* as 3 and *Instantiate Dist* as 4_1, and shifts attention to *Instantiate Dist* with MSN 4_1 because the previous plan *Infer Near* (to understand *near* with context information) fails. Similarly, by applying **EAA4_1**, the GIS agent finds another appropriate method, the recipe, *Ask*, of *Negotiate* (Figure **A-12** (b)), to instantiate the parameter *Dist* based on the previous distance value instantiated.

After receiving the GIS agent's question in G (2) "do you want to increase or decrease the spatial distance *near*?", the user replies to this question with U (3) "decrease, please" because he does not want to go to farther grocery stores due to frequent shopping needs. By following the user's suggestion in U (3), the GIS agent decreases the *near* distance from 5 miles to 2.5 miles, and re-generates responses based on the new distance in G (3): "Here are the grocery stores within 2.5 mile of Apartment A. Is this OK?". The current *PlanGraph* around the plan *Ask* of *Negotiate* is shown in Figure **6-31**. The map response in G (3) is shown in Figure **6-32**.



For missed comopnents, please refer to those in Figure **6-28**

Figure **6-31**: *PlanGraph* after G (3) in **Scenario 4**

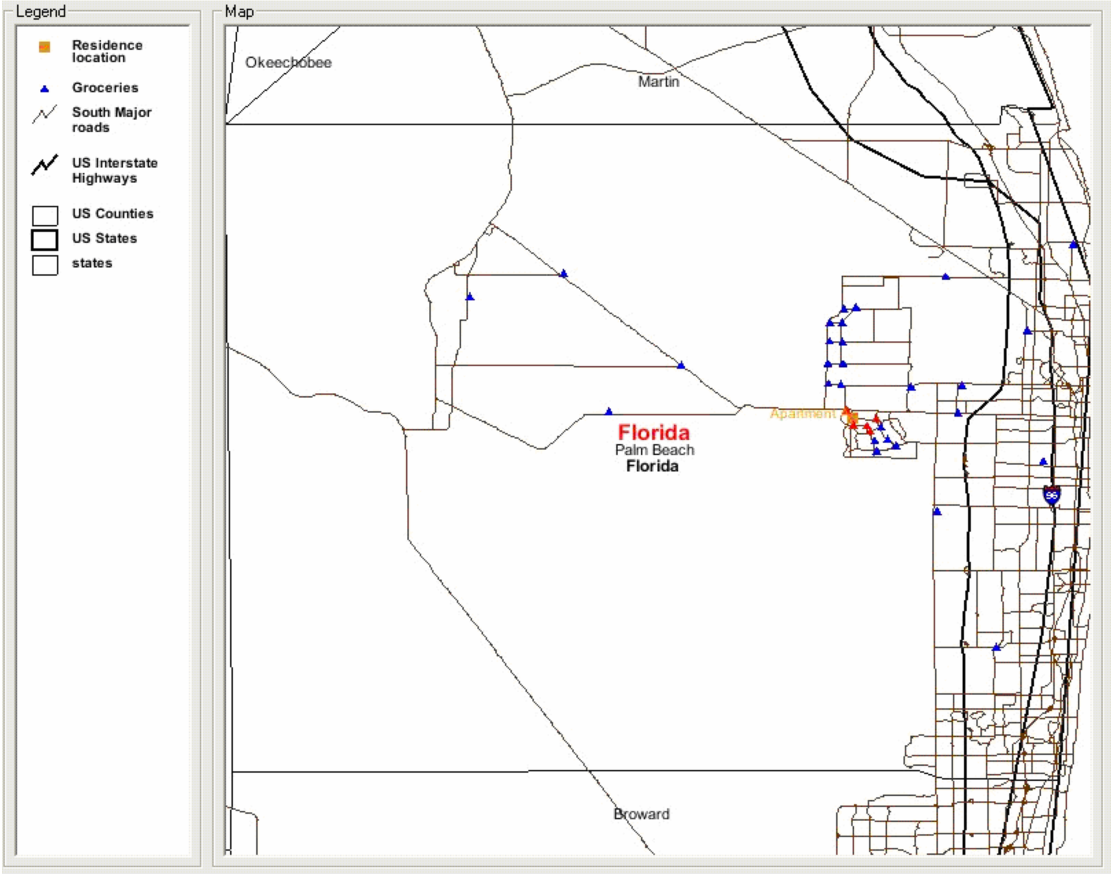


Figure **6-32**: Map Response in G (3) in **Scenario 4**

The user is not satisfied with the grocery stores shown in G (3). From his several traveling experiences to the two grocery stores within 3 miles of Apartment A, he knows that the traffic status around these two stores is acceptable. In addition, from the existing GIS map, he finds that relative spatial distances of these two stores with respect to all grocery stores in this area are not too far. Therefore, he believes that these two stores should also be considered as *near*. Similar to **Scenario 3**, the user also gives his understanding about *near* with a gesture directly in U (4): “No. Please highlight all these groceries.” With the same reasoning process as that in **Scenario 4** for processing U (4), the GIS agent re-generates the map response based on the user’s gesture direction in G (5). Finally, the user’s positive response in U (5) confirms that the user and the GIS agent have reached a successful communication about *near*. The final map response in G (4) is shown in Figure **6-33**.

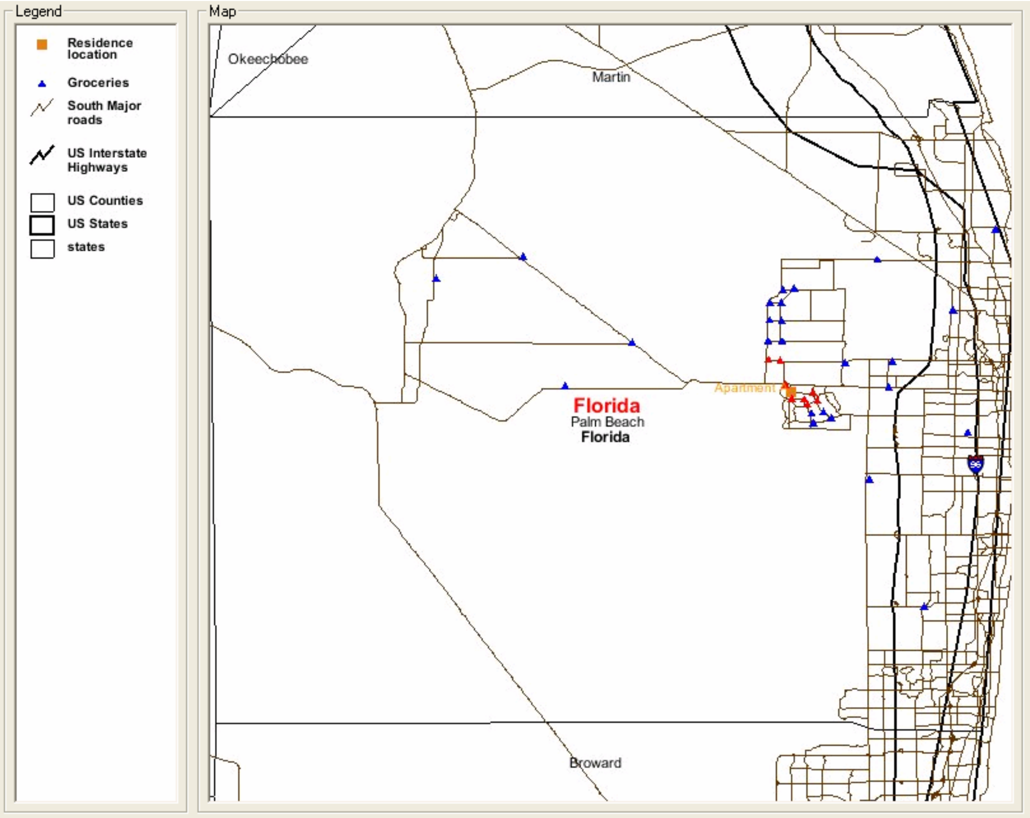


Figure **6-33**: Map Response in G (4) in **Scenario 4**

The major difference between this scenario and the previous three scenarios is that, in this scenario, the two agents do not reach a shared understanding about the current context behind the user request involving the vague spatial concept, while they do in the last three scenarios. This is because in this scenario, the user holds certain context information, but does not share it with the system. If the user actively shares such information with the system, they would be able to build a shared context. In this case, the two agents can still reach a common understanding about the vague spatial concept through collaborative human-GIS dialogues although they do not reach a shared understanding about the current context. At first, collaborative context-sharing dialogues (including not only utterances, but also GIS maps and gestures) help them to ground their understandings about the current context to be close, although not common. Then, other types of collaborative dialogues, e. g. negotiation and direction communication dialogues, can help to further reduce vagueness until the two agents reach a common understanding about the vague spatial concept involved.

6.6 Summary and Discussion

This section summarizes various situations of the context-dependency problem in human-GIS communication and describes how the *PlanGraph* model can be extended to handle the context-dependency problem in four sample human-GIS dialogues. These sample dialogues demonstrate that the collaborative dialogue approach and the *PlanGraph* model can enable the GIS agent to have the flexibility of handling the context-dependency problem in various situations.

These sample human-GIS dialogues demonstrate how the collaborative dialogue and the *PlanGraph* model can help the GIS agent to handle the context-dependency problem in various ways. Firstly, they can help the GIS agent to maintain knowledge on the dynamic context in human-GIS communication. At the beginning of these scenarios, the *PlanGraph* is empty. As the dialogue proceeds, it records and updates contextual information exchanged between two agents and contextual information inferred by the GIS agent. As a result, it enables the GIS agent to keep track of the dynamic context involved in human-GIS dialogues. Secondly, they enable the GIS agent and the user to build a shared context in various situations through various collaborative context-sharing dialogue strategies. With *PlanGraph* keeping track of the context involved, the GIS agent knows what contextual information has not been shared between the two agents. Therefore, it can initiate context-sharing dialogues to ask the user for such information (see **Scenario 1** and **2**) or actively share such information with the user (see **Scenario 3**). This model also enables the GIS agent to understand the user's context-sharing dialogues to ask for (**Scenario 3**) or actively share certain contextual information with the system (see **Scenario 1**, **2**, **3**, and **4**). Thirdly, they enable the GIS agent to understand the meaning of the vague spatial concept with dynamic context information. These sample dialogues demonstrate that the GIS agent can retrieve contextual information from the shared context initially represented by the *PlanGraph*, and then understand the meaning of the vague spatial concept with the retrieved contextual information based on its model of the meaning of the concept. Fourthly, they can help the GIS agent and the user to reach a shared understanding about the meaning of a vague spatial concept. The shared context built in human-GIS communication constrains the GIS agent and the user's understandings about the vague spatial concept to be common or closer. These sample dialogues show that, if needed, the GIS and the user can further ground their understandings about that concept into a common understanding through various collaborative human-GIS dialogue strategies, e. g. negotiation (see **Scenario 4**) or direct communication (see **Scenario 3**, and **4**). Finally, they enable the GIS to handle the context-dependency problem in various situations where the system and the user have different understandings about the current context. The context-dependency problems in the four sample dialogues are in four different representative situations defined in this study. The GIS agent successfully reaches a successful communication of a vague spatial concept with the user in these four dialogues.

The performance of *GeoDialogue* is limited by the limited knowledge built into the *PlanGraph* model. In the sample dialogues described in this chapter, the GIS agent's model about the meaning of the vague spatial concept *near* involves only seven fixed major contextual factors, and does not include certain contextual factors commonly considered in the human's model, e. g. the relative distance contextual factor, time and financial budget. In addition, *GeoDialogue* can not adjust its model when it knows that the user considers more contextual influences (see **Scenario 3** and **4**). If we could build a better model of the meaning of *near* (e. g. including more contextual factors that the human user considers), and better reasoning algorithms (e. g. to enable the GIS agent to adjust its model of the meaning of the vague spatial concept along with the change of the context structure in the *PlanGraph*) for the *PlanGraph* model, *GeoDialogue* would be able to give a better inference on the meaning of a vague spatial concept. Such improvement of the *PlanGraph* model would ease human-GIS communication.

Chapter 7

Conclusion and Future Work

In this study, a collaborative dialogue approach is proposed and developed for spoken natural language-enabled GIS to handle the context-dependency problem involved in human-GIS communication. The Human Communication Framework (HCF) is developed in this study for us to better understand and handle the vagueness problem involved in human-GIS communication by referring to the vagueness problem in human-human communication. This framework directs us to propose the collaborative dialogue approach in this study. The agent-based computational model, the *PlanGraph* model, is developed for us to implement the collaborative dialogue approach. We have also implemented the collaborative dialogue approach and the *PlanGraph* model in the prototype software agent, *GeoDialogue*. The sample dialogues between *GeoDialogue* and the user demonstrate that the *PlanGraph* model and the collaborative dialogue approach can help the GIS to effectively handle the context-dependency problem in various situations.

The remainder of this section discusses contributions made by this study, comparisons between this study and previous studies, and future research, respectively.

7.1 Contributions

The goal of this study is to facilitate the development of natural multimodal GIS, in particular facilitate the handling of the vagueness problem in human-GIS communication. This study has made the following two major contributions toward this goal.

Human Communication Framework

The first major contribution is Human Communication Framework (HCF). This framework plays four major roles in this study as follows.

Firstly, it facilitates our comprehensive understanding about the vagueness problem in human-GIS communication. The sources of vagueness in human-human communication facilitate our understanding about major origins of vagueness in human-GIS communication of spatial concepts. The five major dimensions of the shared context in human-human communication help us to understand the three major types of contextual factors, and the distributed nature of context knowledge in human-GIS communication. The distributed nature of context knowledge in human-GIS communication leads to the need for building a shared context in human-GIS communication and the need for human-GIS collaborative dialogues.

Secondly, this framework helps us to understand various situations of the context-dependency problem in human-GIS communication. The HCF indicates that communicators may have different understandings about the current context in various situations. We have classified these various situations in human-GIS communication into five different categories of situations, where the GIS agent and the user understands the context 1) with different concepts referring to the context, 2) with different values of partial components in the context structure, 3) different widths or 4) different depths of partial components of the context structure, and 5) with a combination of the problems in the first four categories of situations.

Thirdly, this framework indicates success and effectiveness of human solutions involved in handling the vagueness problem. The success of human solutions, that is, collaborative human-human dialogues, motivated us to propose and develop the collaborative dialogue approach for handling the context-dependency problem in human-GIS communication.

Finally, this framework also guides our design and development of the collaborative dialogue approach and the *PlanGraph* model. The major design principle of the collaborative dialogue approach is to enable the GIS to handle the vagueness problem by emulating a human agent's role derived from the HCF. The *PlanGraph* model is and can be further extended by following the HCPs in the HCF.

Collaborative Dialogue Approach

The second major contribution of this study is the collaborative dialogue approach. This study provides a conceptual framework of the collaborative dialogue approach. Section 4.2 gives conceptual description about design of this approach. The major design principle of this approach is to enable human-GIS collaborative dialogues for the system for the GIS to handle the context-dependency problem by emulating human-human collaborative dialogues. During the collaborative human-GIS planning process, the system needs to keep track of the shared context in human-GIS communication, and repair the shared context and further reduce vagueness through collaborative human-GIS dialogues.

This study also offers the *PlanGraph* model to support the collaborative dialogue approach. This model plays several major roles in the GIS agent's handling the context-dependency problem as follows. Firstly, this model enables the GIS agent to maintain dynamic knowledge on the shared context in human-GIS communication, including the task context, the spatial context, and the human context, which influence the meaning of the vague spatial concept. Secondly, this model enables the GIS agent and the user to reach a shared understanding about the context, or a closer understanding about the context if not fully shared. Thirdly, this model enables the GIS agent to understand the meaning of a vague spatial concept under constraints of the shared context built in human-GIS communication. Fourthly, this model enables the GIS agent to reach a shared understanding about the meaning of the vague spatial concept with the user. Finally, this model enables effective human-GIS communication of vague spatial concepts in various situations.

This study also demonstrates how the collaborative dialogue approach and the *PlanGraph* model can be implemented through the prototype software agent, *GeoDialogue*. The sample dialogues between *GeoDialogue* and human users demonstrate the success and effectiveness of this approach and this model.

7.2 Comparison with Previous Studies

As reviewed in Chapter 2, numerous previous studies have been performed to handle the vagueness problem in human-GIS communication. Most of them focus on handling the fuzziness sub problem, and their results support the system to work in only fixed contexts. Among these previous studies, only a few studies, e. g. studies by Wang (1994, 2000, 2003), and Yao and Thill (2006), have addressed how to handle the context-dependency problem in human-GIS communication. Wang (1994) gave several examples about how the system can vary its inference on the meaning of a vague spatial concept depending on contextual information in a user request. Yao and Thill (2006) provide a conceptual framework about how the system can perform qualitative reasoning with context-contingent models of qualitative spatial concepts. The remainder of this section focuses on comparison between our study and these two attempts (Table **7-1**).

Table **7-1**: Comparison with Previous Studies

Capability of the GIS enabled	This study	Wang's study	Yao and Thill's study
Maintaining dynamic context knowledge	Yes	No	No
Building a shared context with the user	Yes	May not	May not
Reaching a shared meaning with the user	Yes	May not	May not
Having flexibility of handling the problem	Yes, in various situations	Yes, but in only limited situations	Yes, but in limited situations

Capability to Keep Track of the Dynamic Context

This study enables the GIS to keep track of the dynamic context involved in human-GIS communication, while Wang's studies and Yao and Thill's study do not address how the dynamic context is modeled in the system. The collaborative dialogue approach and the *PlanGraph* model developed in this study enable the GIS agent to maintain dynamic knowledge on the dynamic context as human-GIS communication evolves. The dynamic context-knowledge facilitates the GIS's understanding about not only the vague spatial concept under communication, but also other concepts in the same context. In Wang's studies, it is not clear how and where the context knowledge is modeled. In the conceptual framework developed by Yao and Thill, context knowledge is directly encoded in context-dependent models of specific vague spatial concepts. With the use of this framework, it is hard for the system to retrieve the context knowledge for later use if needed.

This study enables the GIS agent to keep comprehensive context knowledge similar to human context knowledge, but the studies by Wang and Yao and Thill do not. This study enables the system to keep track of the three major types of contextual factors that influence the meaning of a vague spatial concept. The recursive structure of the context knowledge kept in the *PlanGraph* emulates that of human context knowledge. Wang's studies can enable the GIS to keep knowledge on only part of these three contextual factors. In his studies, it is not clear how to organize the contextual factor in the system's knowledge. With the conceptual framework provided by Yao and Thill's study, the system may keep knowledge on varieties of contextual factors considered in the system's context-dependent model of the meaning of a vague spatial concept. However, it is not clear how these factors are organized in the system's knowledge.

Capability to Build a Shared Context

This study enables the GIS and the user to reach a shared understanding about the dynamic context involved in human-GIS communication, while the system with approached developed by Wang and Yao and Thill may not be able to reach a shared context with the user. The collaborative dialogue capabilities provided by the collaborative dialogue approach and the *PlanGraph* model enable the GIS to build a shared context with the user through collaborative human-GIS dialogues. With the approach in Wang's studies, the system acquires context information directly from the user's spatial information request. As a result, the context information not specified in the user's request may not be shared between the GIS and the user. With the conceptual framework in Yao and Thill's study, the GIS can actively request context information from the user needed in the system's context-dependent model of the meaning of a vague spatial concept. However, the system and the user do not communicate for other contextual information, e. g. that in the user's context-dependent model of the meaning of that concept. If the user considers different contextual factors from those considered by the system, the context may not be shared between the GIS and the user.

Capability to Reach a Shared Meaning

This study enables the GIS agent and the user to reach a shared understanding about the meaning of a vague spatial concept, but the previous studies do not. The shared context enabled by the collaborative dialogue approach and the *PlanGraph* model can constrain the two agents' understandings about the vague spatial concept to be close or common. Collaborative dialogue capabilities enabled by this approach and this model can help the two agents to further ground their understandings into a common one. The approach in Wang's studies supports the GIS to keep only limited dynamic context information, which limits the capability of the GIS to understand the meaning of a vague spatial concept. With the conceptual framework in Yao and Thill's study, the system can give only an estimated meaning of a vague spatial concept based on contextual information considered by itself, however, the user may understand that concept with different contextual information. Therefore, both Wang's studies and Yao and Thill's studies can enable the system to provide only a best guess to the meaning of a vague spatial concept involved in communication, which may not be agreed on by the user in communication.

Flexibility to Handle the Context-Dependency Problem in Various Situations

This study enables the GIS to have the flexibility to handle the context-dependency problem in various situations. However, the results of previous studies can support the GIS to handle the context-dependency problem in only limited situations. This study enables the system to reach a successful communication about a vague spatial concept with different users for different tasks in different spatial contexts in various situations where the system and the user have different understandings about the current context. With Wang's approach, the system can handle the context-dependency problem only in a specific situation in **Situation B**, where the system and the user have different understandings on only the major contextual factor considered by the system. With Yao and Thill's conceptual framework, the system may be able to handle the vagueness problem in **Situation A** and **Situation B**, if the system and the user reach a shared context. If the context-dependency problem is in **Situation C**, **D,** and **E**, that is, where the user and the system consider different contextual influences, the system with this framework may not reach a shared context with the user, which may lead to their different understandings about a vague spatial concept in communication.

7.3 Future Directions

This study has taken a step toward the challenge of handling the context-dependency problem that must be met in development of natural multimodal GIS. It can be further extended in five directions, including a knowledge elicitation study, an evaluation study, extension for handling other communication problems, extension for management of human-GIS communication in various computing environments, and extension for multimodal information fusion in natural multimodal GIS interfaces. We discuss these directions in the remaining section.

Knowledge Elicitation Study

A knowledge elicitation study (Cooke, 1994) that collects human intelligence involved in handling the context-dependency problem would help to design and implement a more practical *PlanGraph* model for the spoken natural language enabled GIS. The performance of the GIS agent with the *PlanGraph* model depends on the human intelligence built into the system, in particular that built into the *PlanGraph* model. In the current implementation of *GeoDialogue*, the general knowledge of actions and recipes, mental states, and reasoning algorithms in the *PlanGraph* model are designed and implemented based on general HCPs from the HCF, not from the human knowledge elicited from human GIS operators/users in various application domains. Therefore, in the future, a more practical *PlanGraph* model based on results of a knowledge elicitation study would further improve the dialogue agent's performance to handle the context-dependency problem in human-GIS communication.

Evaluation Study

The sample dialogues described in Chapter 6 demonstrate the effectiveness of the *PlanGraph* model and the collaborative dialogue approach for the GIS to handle the context-dependency problem. However, if we need to know how well they work, an evaluation study is necessary. An evaluation study can be designed based on comparison between users' performance with using a GIS with the *PlanGraph* model and that with using a GIS with other approaches.

Extension for Handling Other Communication Problems

The collaborative dialogue approach and the *PlanGraph* model can be extended to help the natural multimodal GIS to handle other communication problems, e. g. the incompletion problem, and the ambiguity problem. With the *PlanGraph* keeping track of the dynamic context involved in human-GIS communication, the GIS would be able to make a better inference on the missing information or the meaning of an ambiguous input in the user's request under constraints of the current context. With the collaborative dialogue abilities provided by the *PlanGraph* model, the GIS could further reduce uncertainties involved in communication through collaborative dialogues with the user.

Extension for Management of Human-GIS Communication in Various Computing Environments

The collaborative dialogue approach and the *PlanGraph* model can be extended to facilitate human-GIS communication in various computing environments, e. g. GIS in PDA, and CSCW. In this study, we consider only one human user involved in human-communication, while the natural multimodal GIS may need to communicate with multiple people, e. g. in CSCW. This approach and this model can be extended to keep track of the dynamic context involved in human-GIS-human communication, and facilitate handling communication problems involved in such communication. Currently, we consider design of the *PlanGraph* model only for its application in GIS used in a general computer platform, e. g. a desktop PC or a PC with a large screen display. When human-GIS communication involves the use of a different computer platform, e. g. PDA, the context involved in communication between the user and such a computer system may have different characteristics, e. g. the human context may be fixed, and the spatial context may not be shared as much as that in communication between the user and a system with a large screen display. The *PlanGraph* model needs to be further extended to model the shared context involved in such human-GIS communication.

Extension for Multimodal Information Fusion in Natural Multimodal GIS Interfaces

The *PlanGraph* model can be extended to facilitate multimodal information fusion in natural multimodal GIS. In the natural multimodal GIS, the multimodal input from the user may include multiple segments pointing to the same object, or different targets (Sharma et al., 1998). With the *PlanGraph* model keeping track of the context involved in human-GIS communication, the system would be able to constrain its interpretation of each segment in the multimodal input. For example, the system could infer the meaning of each segment based on its contributions and relevance to the user's intention and the human-GIS collaboration represented in the *PlanGraph*. Furthermore, the system could determine which segments represent the same meaning, and which segments complement each other.

Bibliography

Ahlqvist, O. (2000). *Context sensitive transformation of geographic information.* Unpublished Ph. D Thesis, Stockholm University.

Ahlqvist, O., Keukelaar, J., & Oukbir, K. (2000). Rough classification and accuracy assessment. *Journal of Geographical Information Science, 14*, 475-496.

Balkanski, C., & Hurault-Plantet, M. (2000). Cooperative requests and replies in a collaborative dialogue model. *International Journal of Human-Computer Studies, 53*(6), 915-968.

Barsalou, L. W. (1982). Context-independent and context-dependent information in concepts. *Memory & Cognition, 10*(1), 82-93.

Barsalou, L. W. (1987). The instability of graded structure: Implications for the nature of concepts. In U. Neisser (Ed.), *Concepts and conceptual development: Ecological and intellectual factors in categorization* (pp. 101-140). Cambridge: Cambridge University Press.

Barsalou, L. W. (1999). Perceptual symbol systems. *Behavioral and Brain Sciences, 22*, 577-609.

Barsalou, L. W. (2000). Concepts: Structure. In A. E. Kazdin (Ed.), *Encyclopedia of Psychology* (Vol. 2, pp. 245-248). New York: Oxford University Press.

Barsalou, L. W. (2003). Situated simulation in the human conceptual system. *Language and Cognitive Processes, 18*, 513-562.

Barsalou, L. W., & Hale, C. R. (1993). *Components of conceptual representation: From feature lists to recursive frames.* Paper presented at the Categories and concepts: Theoretical views and inductive data analysis, San Diego, CA.

Barsalou, L. W., & Hutchinson, J. W. (1987). *Schema-based planning of events in consumer contexts.* Paper presented at the Advances in Consumer Research, Provo, UT.

Berger, C. R. (1975). Proactive and retroactive attribution processes in interpresonal communications. *Human communication research, 2*(1), 33-50.

Berger, C. R. (1987). Communicating under uncertainty. In M. E. Roloff & G. R. Miller (Eds.), *Interpersonal processes: New directions in communication research* (pp. 39-62). Newbury Park, CA: Sage.

Berger, C. R., & Calabrese, R. J. (1975). Some explorations in initial interaction and beyond: Toward a developmental theory of interpersonal communication. *Human communication research*(1), 99-112.

Bolt, R. A. (1980). *Put-that-there: voice and gesture at the graphics interface.* Paper presented at the Proceedings of the 7th annual conference on computer graphics and interactive techniques, Seattle, Washington, United States.

Bosc, P., & Prade, H. (1993). *An introduction to fuzzy set and possibility theory based approaches to the treatment of uncertainty and imprecision in database management systems.* Paper presented at the Proceedings of the 2nd Workshop on Uncertainty Management in Information Systems: From Needs to Solutions, Catalina, CA.

Bratman, M. E. (1987). *Intentions, Plans, and Practical Reason*. Cambridge, MA: Harvard University Press.

Bratman, M. E. (1992). Shared cooperative activity. *Philosophical Review, 101*, 327-341.

Briggs, R. (1973). Urban Cognitive Distance. In R. M. Downs & D. Stea (Eds.), *Image and Environment: Cognitive mapping and spatial behavior* (pp. 361-388). Chicago: Aldline Publishing Company.

Brodeur, J., Bédard, Y., Edwards, G., & Moulin, B. (2003). Revisiting the Concept of Geospatial Data Interoperability within the Scope of Human Communication Processes. *Transactions in GIS, 7*(2), 243-265.

Burrough, P. A., & Frank, A. U. (1996). *Geographic Objects with Indeterminate Boundaries*. London: Taylor & Francis.

Burrough, P. A., Gaans, P. F. M. v., & MacMillan, R. A. (2000). High-resolution landform classification using fuzzy k-means. *Fuzzy Sets and Systems, 113*(1), 37-52.

Cai, G., Wang, H., & MacEachren, A. M. (2003, September 24-28). *Communicating Vague Spatial Concepts in Human-GIS Interactions: A Collaborative Dialogue Approach.* Paper presented at the Conference on Spatial Information Theory 2003, Kartause Ittingen, Switzerland.

Cai, G., Wang, H., MacEachren, A. M., & Sven, F. (2005). Natural Conversational Interfaces to Geospatial Databases. *Transactions in GIS, 9*(2), 199-221.

Cai, G., & Xue, Y. (2006, January 29 – February 1, 2006). *Activity-oriented Context-aware Adaptation Assisting Mobile Geo-spatial Activities.* Paper presented at the ACM IUI’2006 – International Conference on Intelligent User Interface, Sydney, Australia.

Carey, S. (1985). *Conceptual change in childhood*: MIT Press.

Cherry, C. (1978). *On human communication: a review, a survey, and a criticism* (Third ed.). Cambridge, Massachusetts: The MIT Press.

Chuah, M. C., & Roth, S. F. (2003, 16-18 July 2003). *Visualizing common ground.* Paper presented at the Seventh International Conference on Information Visualization.

Clark, H. (1996). *Using language*. Cambridge: Cambridge University Press.

Cohen, P., Dalrymple, M., Moran, D., & Pereira, F. (1989). *Synergistic use of direct manipulation and natural language.* Paper presented at the CHI '89.

Cohen, P. R., & Levesque, H. J. (1991). Teamwork. *Nous, 25*(4), 487-512.

Cohn, A. G., & Gotts, N. M. (1996). The 'Egg-Yolk' Representation of Regions with Indeterminate Boundaries. In *Geographic Objects with Indeterminate Boundaries, GISDATA Series* (Vol. 3, pp. 171-187): Taylor & Francis.

Cooke, N. J. (1994). Varieties of knowledge elicitation techniques. *International Journal of Human-Computer Studies, 41*, 801-849.

Couclelis, H. (1996). Towards an Operational Typology of Geographic Entities with Ill-defined Boundaries. In P. A. Burrough & A. V. Frank (Eds.), *Geographic Objects with Indeterminate Boundaries (GISDATA Series)* (Vol. 3, pp. 45-55). Bristol, PA: Taylor & Francis.

Cross, V., & Firat, A. (2000). Fuzzy objects for geographical information systems. *Fuzzy Sets and Systems, 113*(1), 19-36.

Day, R. A. (1976). Urban distance cognition: review and contribution. *Australian Geographer, 13*, 193-200.

Dimitrov, V., & Russell, D. (1994). The Fuzziness of Communication. In L. Fell, D. Russell & A. Stewart (Eds.), *Seized by Agreement, Swamped by Understanding: Acollection of papers to celebrate the visit to Australia, in August 1994* (pp. 183-192). Sydney, Australia: Hawkesbury Printing, University of Western Sydney.

Duckham, M., Mason, K., Stell, J., & Worboys, M. F. (2001). A formal approach to imperfection in geographic information. *Computers, Environments and Urban Systems, 25*, 89-103.

Egenhofer, M. J. (1996). Multi-modal spatial querying. In J. M. Kraak & M. Molenaar (Eds.), *Advances in GIS Research II (Proceedings of The Seventh International Symposium on Spatial Data Handling)* (pp. 785-799). London: Taylor & Francis.

Egenhofer, M. J., & Kuhn, W. (1999). Interact with GIS. In P. A. Longley, M. F. Goodchild, D. J. Maguire & D. W. Rhind (Eds.), *Geographical Information Systems* (Vol. 1, pp. 401-402). New York: John wiley & Sonsm Inc.

Egenhofer, M. J., & Mark, D. M. (1995, September 1995). *Naive Geography.* Paper presented at the COSIT `95, Semmering, Austria.

Egenhofer, M. J., & Shariff, A. R. (1998). Metric details for natural-language spatial relations. *ACM Transactions on Information Systems, 16*(4), 295 - 321.

Erwig, M., & Schneider, M. (1997). *Vague regions.* Paper presented at the Advances in spatial databases(SSD '97), Berlin.

ESRI. (2002). *ArcXML Programmer's Reference Guide - ArcIMS 4.0.* Redlands, CA: ESRI.

Fillmore, C. J. (1985). Frames and the semantics of understanding. *Quaderni di Semantica, 6*, 222-255.

Fischer, G., Lemke, A. C., Mastaglio, T., & Morch, A. I. (1991). The role of critiquing in cooperative problem solving. *ACM Transactions on Information Systems, 9*(3), 123-151.

Fisher, P. (2000). Sorites paradox and vague geographies. *Fuzzy Sets and Systems, 113*, 7–18.

Fisher, P., & Orf, T. (1991). An investigation of the meaning of near and close on a university campus. *Computers, Environment and Urban Systems, 15*, 23–25.

Florence, J., Hornsby, K., & Egenhofer, M. J. (1996). *The GIS Wallboard: interactions with spatial infor-mation on large-scale display.* Paper presented at the Seventh International Symposium on Spatial Data Handling (SDH '96), Delft, The Netherlands.

Foote, K. E., & Lynch, M. (1995, Oct. 12, 1997). *Geographic Information Systems as an Integrating Technology: Context, Concepts, and Definitions*. Retrieved Apr. 27, 2004, 2004, from http://www.colorado.edu/geography/gcraft/notes/intro/intro.html

Frank, A. U. (1993). The Use of Geographical Information Systems: The User Interface is the System. In D. Medyckyj-Scott & H. M. Hearnshaw (Eds.), *Human Factors in Geographical Information Systems* (pp. 3-14). London: Belhaven Press.

Frank, A. U., & Mark, D. M. (1991). Language issues for GIS. In D. Macguire, M. F. Goodchild & D. Rhind (Eds.), *Geographical Information Systems: Principles and Applications* (pp. 147-163). New York: Wiley.

Gahegan, M. (1995). *Proximity operators for qualitative spatial reasoning.* Paper presented at the Spatial Information Theory: A Theoretical Basis for GIS, Berlin.

Galton, A. (2003). Granularity-sensitive Spatial Attributes. *Spatial Cognition and Computation, 3*(2/3), 97 - 118.

Golledge, R. G. (1991). Environmental cognition. In D. Stokols & I. Altman (Eds.), *Handbook of Envrionmental Psychology*. Malabar, FL: Krieger Publichsing Company.

Golledge, R. G. (2002). The nature of geographical knowledge. *Annals of the Association of American Geographers, 92*(1), 1-14.

Grosz, B. J., & Kraus, S. (1996). Collaborative plans for complex group action. *Artificial Intelligence, 86*(2), 269-357.

Grosz, B. J., & Kraus, S. (1999). The Evolution of SharedPlans. In A. Rao and M. Wooldridge (Ed.), *Foundations and Theories of Rational Agency* (pp. 227-262).

Grosz, B. J., & Sidner, C. L. (1986). Attention, intentions, and the structure of discourse. *Computational Linguistics, 12*(3), 175--204.

Grosz, B. J., & Sidner, C. L. (1990). Plans for discourse. In P. R. COHEN, MORGAN, J. L. and POLLACK, M. E. (Ed.), *Intentions in Communication* (pp. 417-444). Cambridge, MA: MIT Press.

Guesgen, H. W., & Albrecht, J. (2000). Imprecise reasoning in geographic information systems. *Fuzzy Sets and Systems, 113*(1), 121-131.

Head, C. G. (1991). Mapping as language or semiotic system: review and comment. In D. M. Mark & A. U. Frank (Eds.), *Cognitive and L inguistic Aspects of Geographic Space* (pp. 237-262). Dordrecht: Kluwer.

Heath, R. L., & Bryant, J. (2000). *Human communication theory and research: concepts, contexts, and challenges* (2 ed.). Mahwah, New Jersey: Lawrenece Erlbaum Associates, Publisher.

Hernández, D., E, C., & D., F. P. (1995). *Qualitative distances.* Paper presented at the Spatial Information Theory: A Theoretical Basis for GIS.

Hirtle, S. C., & Hudson, J. (1991). Acquisition of spatial knowledge for routes. *Journal of Environmental Psychology, 11*, 335- 345.

Hyde, D. (2005). *Definition of Sorites Paradox*. Retrieved Aug. 21, 2006, 2006, from http://plato.stanford.edu/entries/sorites-paradox/

Jackendoff, R. S. (1991). Parts and boundaries. *Cognition, 41*, 9-44.

Jebara, T. (2004). *Machine Learning: Discriminative and Generative*. Boston: Kluwer Academic Publishers.

Kang, S. (1997). Factors to Consider: Developing Adult EFL Students' Speaking Abilities. *Forum, 35*(3), 8.

Kaplan, S. (1973). Cognitive maps in perception and thought. In R. M. Downs & D. Stea (Eds.), *Image and Environment: Cognitive mapping and spatial behavior* (pp. 63-78). Chicago: Aldline Publishing Company.

Keil, F. C. (1989). *Concepts, kinds, and cognitive development*: MIT Press.

Klein, G., Feltovich, P. J., Bradshaw, J. M., & Woods, D. D. (2005). Common Ground and Coordination in Joint Activity. In W. R. Rouse & K. B. Boff (Eds.), *Organizational Simulation*. New York: John Wiley & Sons.

Kollias, V. J., & Voliotis, A. (1991). Fuzzy reasoning in the development of geographical information systems FRSIS: a prototype soil information system with fuzzy retrieval capabilities. *International Journal of Geographical Information Science, 1*(2), 209-223.

Krishnapuram, R., Keller, J. M., & Ma, Y. (1993). Quantitative analysis of properties and spatial relations of fuzzy image regions. *IEEE Transactions on Fuzzy Systems, 1*(3), 222-233.

Kuipers, B. (1978). Modelling Spatial Knowledge. *Cognitive Science, 2*, 129-153.

Kunnapas, T. (1960). Scales for subjective distance. *Scandinavian Journal of Psychology, 1*, 187-192.

Landau, B., & Jackendoff, R. (1993). "What" and "where" in spatial language and spatial congnition. *Behavioral and brain sciences, 16*, 217-265.

Leeds-Hurwitz, W. (1989). *Communication in Everyday Life - a Social Interpretation*. Norwood, New Jersey: Ablex Publishing Corporation.

Lehrer, A. (1974). *Semantic fields and lexical structure*. North-Hottand & New York: American Elsevier.

Leung, Y. (1982). Approximate characterization of some fundamental concepts of spatial analysis. *Geographical Analysis, 14*(1), 29-40.

Leung, Y. (1987). On the imprecision of boundaries. *Geographical Analysis, 19*, 125-151.

Lindsay, P. H., & Norman, D. A. (1977). *Human Information Processing: An Introduction to Psychology*. New York: Academic Press.

Lloyd, R., & Heivly, C. (1987). Systematic distortions in urban cognitive maps. *Annals of the Association of American Geographers, 77*(2), 191-207.

Lochbaum, K. E. (1994). *Using Collaborative Plans to Model the Intentional Structure of Discourse.* Harvard University, Cambridge.

Lochbaum, K. E. (1998). A collaborative planning model of intentional structure. *Computational Linguistics, 24*(4), 525-572.

Lochbaum, K. E., Grosz, B., & Sidner, C. (2000). Discourse structure and intention recognition. In R. Dale, H. Moisl & H. Somer (Eds.), *Handbook of natural language processing* (pp. 123-146). New York: Marcel Dekker, Inc.

Lyotard, J. F. (1984). *The Postmodern Condition: A Report of Knowledge*. Minneapolis: University of Minnesota Press.

MacEachren, A. M. (1995). *How maps work: representation, visualization and design*. New York: Guilford Press.

MacEachren, A. M., & Cai, G. (2006). Supporting Group Work in Crisis Management: Visually mediated human-GIS-human dialogue. *Environment and Planning B: Planning and Design, 33*(3), 435-456.

MacEachren, A. M., Cai, G., Sharma, R., Rauschert, I., Brewer, I., Bolelli, L., et al. (2005). Enabling Collaborative GeoInformation Access and Decision-Making Through a Natural, Multimodal Interface. *International Journal of Geographical Information Science, 19*(3), 293-317.

MacMillan, R. A., Pettapiece, W. W., Nolan, S. C., & Goddard, T. W. (2000). A generic procedure for automatically segmenting landforms into landform elements using DEMs, heuristic rules and fuzzy logic. *Fuzzy Sets and Systems, 113*(1), 81-109.

Malczewski, J. (1999). *GIS and Multicriteria Decision Analysis*. New York: John Wiley & Sons, Inc,.

Mark, D. M. (1993). Human spatial cognition. In D. Medyckyj-Scott & H. M. Hearnshaw (Eds.), *Human Factors in Geographical Information Systems* (pp. 51-60). London: Belhaven Press.

Mark, D. M., & Frank, A. U. (1992a). *NCGIA Initiative 2, "Languages of Spatial Relations," Closing Report.* Santa Barbara, CA: National Center for Geographic Information and Analysis.

Mark, D. M., & Frank, A. U. (1992b). *User interfaces for Geographic Information Systems: report on the specialist meeting* (No. 92-3): National Center for Geographic Information and Analysis.

Mark, D. M., Freksa, C., Hirtle, S. C., Lloyd, R., & Tversky, B. (1999). Coginitive models of geographical space. *International Journal of Geographical Information Systems, 13*(8), 747-774.

Mark, D. M., Svorou, S., & Zubin, D. (1987). *Spatial terms and spatial concepts: Geographic, cognitive, and linguistic perspectives.* Paper presented at the Proceedingsof International Geographic Information Systems (IGIS) Symposium: The Research Adgenda:, Arlington, VA.

McTear, M. F. (2002). Spoken dialogue technology: enabling the conversational user interface. *ACM Computing Surveys, 34*, 90 - 169.

Moar, I., & Carleton, L. R. (1982). Memory for routes. *Quarterly Journal of Experimental Psychology, 34A*, 381± 394.

Montello, D. R., Goodchild, M. F., Gottsegen, J., & Fohl, P. (2003). Where's Downtown?: Behavioral Methods for Determining Referents of Vague Spatial Queries. *Spatial Cognition and Computation, 3*(2/3), 185 - 206.

Neal, J. G., Thielman, C. Y., Funke, D. J., & Byoun, J. S. (1989, Mar. 27 - 31, 1989). *Multi-modal output composition for human-computer dialogues.* Paper presented at the Proceedings of the 1989 AI systems in Government Conference, Washington, D. C.

Nyerges, T. L., Mark, D. M., Laurini, R., & Egenhofer, M. J. (Eds.). (1995). *Cognitive aspects of human-computer interaction for geographic information systems*. Dordrecht, The Netherlands: Kluwer Academic Publishers.

Oviatt, S. L. (1992). *Pen/voice: Complementary multimodal communication.* Paper presented at the Proceeding of Speech Technology'1992.

Oviatt, S. L. (1996). *Multimodal interfaces for dynamic interactive maps.* Paper presented at the Proceedings of the Conference on Human Factors in Computing Systems (CHI'96).

Owei, V. (2002). An intelligent approach to handling imperfect information in concept-based natural language queries. *ACM Transactions on Information Systems, 20*(3), 291-328.

Paivio, A. (1971). *Imagery and Verbal Processes*. New York: Holt, Rinehart & Winston.

Pickering, M. J., & Garrod, S. (2004). Toward a mechanistic psychology of dialogue. *BEHAVIORAL AND BRAIN SCIENCES, 27*, 169–226.

Potter, M. C., & Faulconer, B. A. (1975). Time to Understand Pictures and Words. *Nature, 253*, 437-438.

Presson, C. C., & Montello, D. R. (1988). Points of reference in spatial cognition: Stalking the elusive landmark. *British Journal of Developmental Psychology, 6*, 65-79.

Reisberg, D. (2001). *Cognition: Exploring the Science of the Mind* (second ed.). New York: W. W. Norton & Company.

Robinson, V. B. (1990). Interactive Machine Acquisition of a Fuzzy Spatial Relation. *Computers and Geosciences, 16*(6), 857-872.

Robinson, V. B. (2000). Individual and multipersonal fuzzy spatial relations acquired using human-machine interaction. *Fuzzy Sets and Systems, 113*(1), 133 - 145.

Robinson, V. B., Thongs, D., & Blaze, M. (1985). *Machine acquisition and representation of natural langauge concepts for geographic information retrieval.* Paper presented at the Proceedings of 16th annual Pittsburgh conference.

Schramm, W. (1954). How communication works. In W. Schramm (Ed.), *The process and effects of mass communication* (pp. 3-26). Urbana, IL: University of Illinois Press.

Sharma, J., Flewelling, D. M., & Egenhofer, M. J. (1994). *A qualitative spatial reasoner.* Paper presented at the Proceedings of the Sixth International Symposium on Spatial Data Handling, Edinburgh, Scotland.

Sharma, R., Pavlovic, V. I., & Huang, T. S. (1998). Toward a multimodal human computer interface. *Proceedings of the IEEE, 86*(5), 853-869.

Sharma, R., Yeasin, M., Krahnstoever, N., Rauchert, I., Cai, G., Brewer, I., et al. (2003). Speech-gesture driven multimodal interfaces for crisis management. *Proceedings of the IEEE, 91*(9), 1327 - 1354.

Stell, J. G. (2003). Qualitative Extents for Spatio-Temporal Granularity. *Spatial Cognition and Computation, 3*(2/3), 119 - 136.

Taylor, H. A., & Tversky, B. (1992). Spatial mental models derived from survey and route descriptions. *Journal of Memory and Language, 31*, 261-282.

Terveen, L. G. (1995). Overview of human-computer collaboration. *Knowledge-based Systems, 8*(2-3), 67-81.

Thorndyke, P. W., & Hayes-Roth, B. (1982). Differences in spatial knowledge acquired from maps and navigation. *Cognitive Psychology, 14*, 560-589.

Tubbs, S., & Moss, S. (2000). *Human communication* (Eighth ed.). Boston: McGraw-Hill Higher Education.

Tversky, B. (1981). Distortions in Memory for Maps. *Cognitive Psychology, 13*, 407-433.

Tversky, B. (1993). *Cognitive maps, cognitive collages, and spatial mental models.* Paper presented at the Spatial Information Theory: A Theoretical Basis for GIS, Proceedings Cosit'93.

Varzi, A. C. (2001). Vagueness in Geography. *Philosophy & Geography, 4*(1), 49–65.

Verderber, R. F. (1987). *Communicate!* (5th ed.). Belmont, CA: Wadsworth Publishing Company.

Wang, F. (1994). Towards a natural language user interface: An approach of fuzzy query. *International Journal of Geographical Information Systems, 8*(2), 143-162.

Wang, F. (2000). A fuzzy grammar and possibility theory-based natural language user interface for spatial queries. *Fuzzy Sets and Systems, 113*(1), 147-159.

Wang, F. (2003). Handling Grammatical Errors, Ambiguity and Impreciseness in GIS Natural Language Queries. *Transactions in GIS, 7*(1), 103-121.

Wang, F., & Hall, G. B. (1996). Fuzzy representation of geographical boundaries in GIS. *International Journal of Geographical Information Systems, 10*, 573-590.

Weaver, C. H., & Strausbaugh, W. L. (1964). *Fundamentals of speech communication*. New York: American Book Company.

Wickens, C. D. (1984). *Engineering psychology and human performance*. Columbus, OH: Merrill Publishing Company.

Worboys, M. F. (1998). Imprecision in finite resolution spatial data. *GeoInformatica, 2*(3), 257-280.

Worboys, M. F. (2001a). *Communicating geographic information in context.* Paper presented at the Meeting on Fundamental Questions in GIScience, Manchester.

Worboys, M. F. (2001b). Nearness relations in environmental space. *International Journal of Geographical Information Science, 15*(7), 633–651.

Yao, X., & Thill, J.-C. (2005). How Far Is Too Far? A Statistical Approach to Context-contingent Proximity Modeling. *Transactions in GIS, 9*(2), 157-178.

Yao, X., & Thill, J.-C. (2006). Spatial queries with qualitative locations in spatial information systems. *Computers, Environment and Urban Systems, 30*(4), 485-502.

Zadeh, L. A. (1965). Fuzzy sets. *Information and Control, 8*, 338-353.

Zadeh, L. A. (1978). Fuzzy sets as the basis for a theory of possibility. *Fuzzy Sets and Systems, 1*, 1-27.

Zadeh, L. A. (1983a). Commonsense knowledge representation based on fuzzy logic. *IEEE Computer*(October, 1983), 61-65.

Zadeh, L. A. (1983b). The role of fuzzy logic in the management of uncertainty in expert systems. *Fuzzy Sets and Systems, 11*, 199-227.

Zhan, F. B. (1997). *Topological Relations Between Fuzzy Regions.* Paper presented at the Proceedings of the 1997 ACM Symposium on Applied Computing, San Jose, CA.

Zhan, F. B. (1998). Approximate Analysis of Topological Relations Between Geographic Regions with Indeterminate Boundaries. *Soft Computing, 2*(2), 28-34.

Zhan, F. B. (2002). A Fuzzy Set Model of Approximate Linguistic Terms in Descriptions of Binary Topological Relations between Simple Regions. In P. Matsakis & L. M. Sztandera (Eds.), *Applying Soft Computing in Defining Spatial Relations* (pp. 179-202). Heidelberg, Germany: Physica-Verlag.

Zue, V. W., & Glass, J. R. (2000). Conversational interfaces: Advances and challenges. *Proceedings of the IEEE, 88*, 1166-1180.

Zwaan, R. A., Langston, M. C., & Graesser, A. C. (1995). The construction of situation models in narrative comprehension: An event-indexing model. *Psychological Science, 6*, 292-297.

Zwaan, R. A., & Radvansky, G. A. (1998). Situation models in language comprehension and memory. *Psychological Bulletin, 123*, 162-185.

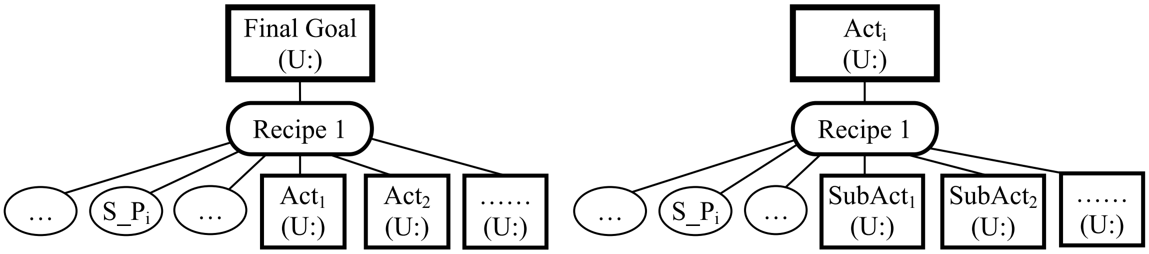
Appendix A

Actions and Recipes in *GeoDialogue*

A.1 Domain Knowledge

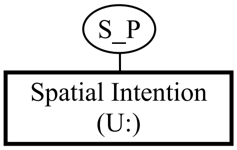
A.1.1 General Contexts Involving User's Spatial Intentions

Here are general actions and recipes involving the user's spatial intentions (Figure **A-1**).



(a) A general recipe for the user's *Final Goal*.
S_P_i Parameter representing a spatial context factor

(b) A general recipe for *Act$_i$* toward *Final Goal*.
S_P_j: Parameter representing a spatial context factor

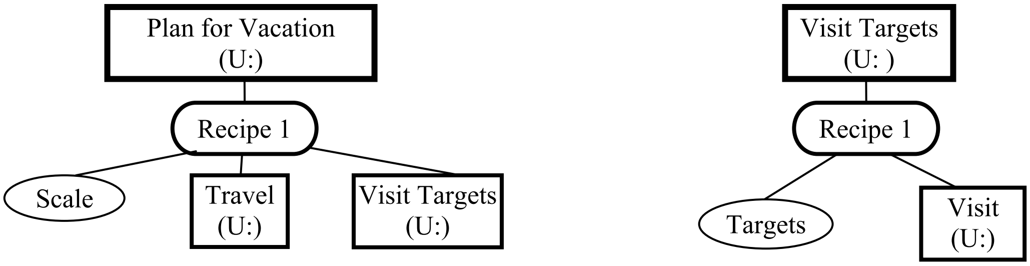


(c) Relation between *Spatial Intention* and *S_P* in actions and recipes toward the user's *Final Goal*.
S_P_i Parameter representing a spatial context factor.

Figure **A-1**: General Actions and Recipes Representing the Context

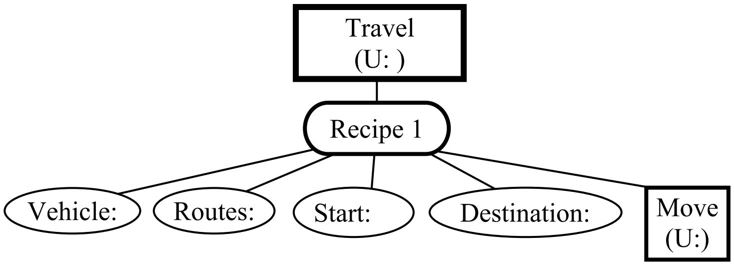
A.1.2 Planning for Vacation

Here are actions and recipes involved in the domain of planning for vacation (Figure **A-2**).



(a) A recipe about how a user achieves the goal *Plan for Vacation*

(b) A recipe about how a user does the sub action *Visit Targets*, where *Targets*: visiting targets



(c) A recipe about how a user achieves his/her goal *Shopping* behind the user's map request to the GIS

Figure **A-2**: Actions and Recipes Involved in Planning for Vacations

A.1.3 Planning for Food Shopping

Here are actions and recipes involved in the domain of planning for food shopping (Figure **A-3**).

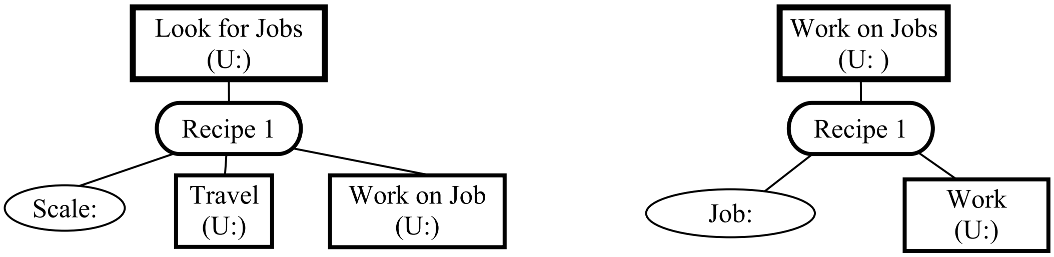
(a) A recipe about how a user achieves his/her goal *Food Shopping*

(b) A recipe about how a user does the sub action *Travel and Shop* involved in *Shopping*

Figure **A-3**: Actions and Recipes Involved in Planning for Food Shopping

A.1.4 Planning for Jobs Looking

Here are actions and recipes involved in the domain of planning for looking for jobs (Figure **A-4**).



(a) A recipe about how a user achieves his/her goal *Look for Jobs*

(b) A recipe about how a user does the sub action *Work* in a location

Figure **A-4**: Actions and Recipes Involved in Planning for Looking for Jobs

A.2 GIS Function Knowledge

A.2.1 Modeling User Intention

Here are all actions and their recipes representing the user intentions (Figure **A-5**).

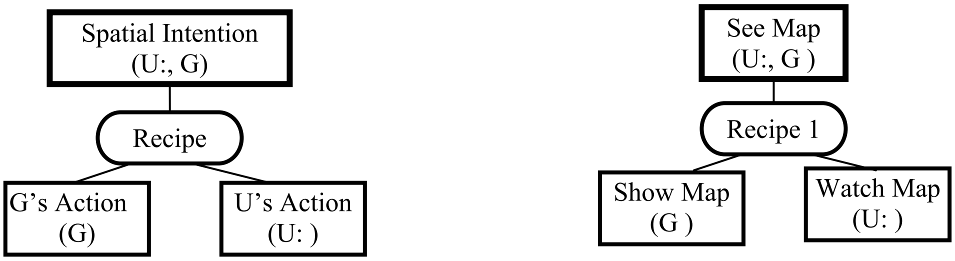
(a) A general action representing the user's intention underlying the user's spatial information request to GIS

(b) An example action, See Map, representing the user's intention, to see a map, underlying the user's map request to GIS

Figure **A-5**: *Spatial Intention* Underlying User's Spatial Information Request

A.2.2 For Helping User's Spatial Information Request

Here are actions and recipes for helping the user's spatial information requests (Figure **A-6**).



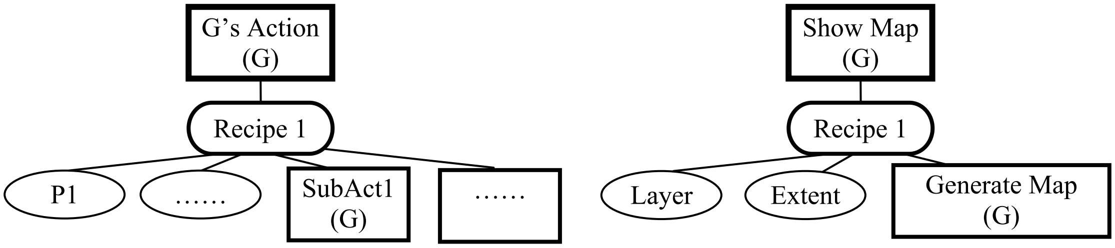
(a) General recipe for GIS to help the user for the user's intention as their common goal

(b) An example recipe for GIS to help the user for the user's intention to see a map

Figure **A-6**: Recipes for GIS to Reach User's Spatial Intention

A.2.3 Modeling GIS Agent's Actions

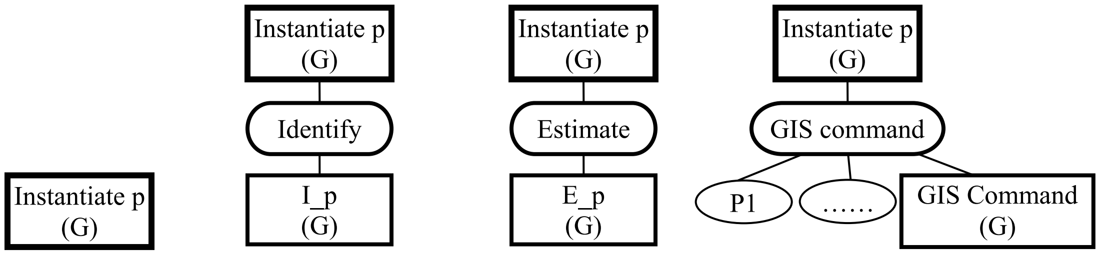
Here are actions and recipes modeling the GIS agent's own goals (Figure **A-7**), and ones for the GIS agent to instantiate associated parameters (Figure **A-8**). Actions and recipes for the GIS agent to understand the meaning of a vague spatial concept (Figure **A-9**), and instantiate parameters representing context information (Figure **A-10**).



(a) General recipe for GIS to execute its action, *G's Action*.
One of SubActs is to execute a GIS command.

(b) An example recipe for the GIS agent's duty, *Show Map*
Layer: the map content
Extent: the map extent

Figure **A-7**: Recipes for *G's Action*

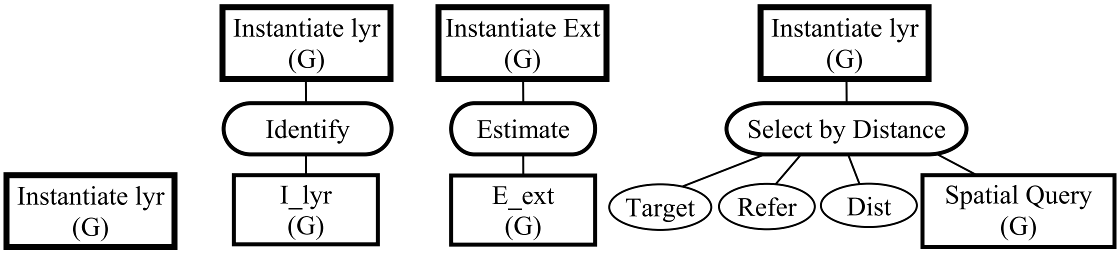


(a) Action for the GIS to instantiate the parameter *p*

(b) One recipe for the GIS to identify the parameter value if it is directly available from the users' request

(c) One recipe for the GIS to estimate the parameter value for *p* if *p* is optional and its value is not available from the user's request

(d) One recipe for the GIS to instantiate the parameter value for *p* by executing a GIS command if *p*'s value must be generated through the GIS command

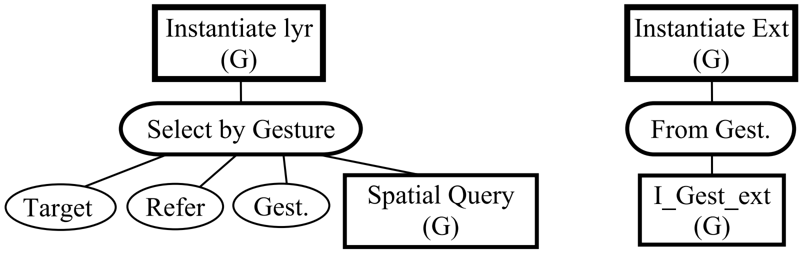


(e) Action for the GIS to instantiate the parameter *Layer*

(f) One recipe for the GIS to identify the layer content

(h) One recipe for the GIS to estimate the map extent *ext*

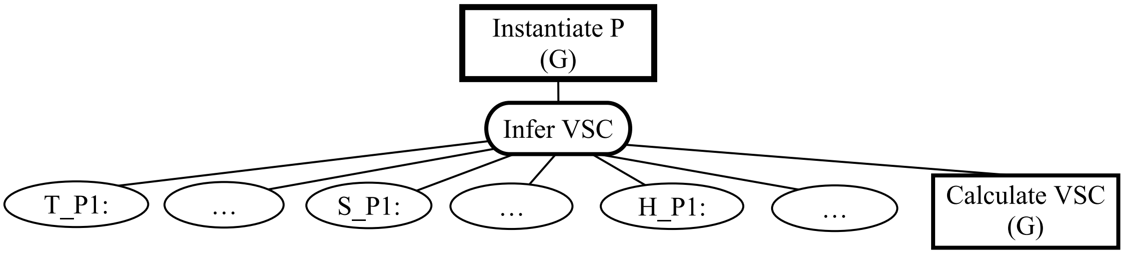
(i) One recipe for the GIS to generate a layer content by executing the GIS command, *Select by Location,* where:
Target: the target map layer;
Refer: the reference location;
Dist: the spatial distance between *Target* and *Refer*



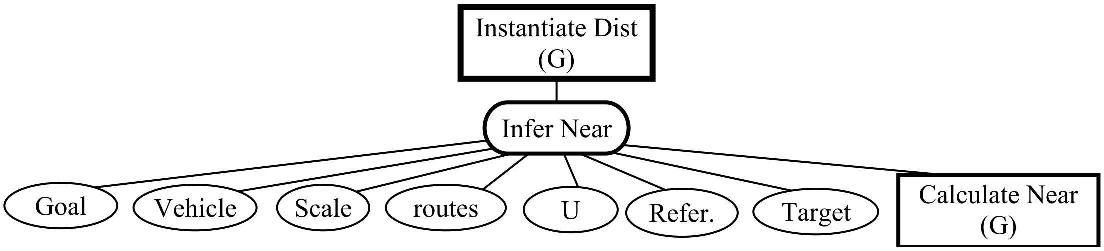
(j) One recipe for the GIS to generate a layer content by executing the GIS command, *Select by Location,* where: *Target*: the target map layer; *Refer*: the reference location; *Gest.*: Gesture by the user

(k) One recipe for the GIS to change the map extent *ext* to the user's gesture area

Figure **A-8**: Action and Recipes for the GIS to Instantiate the Parameter *p*



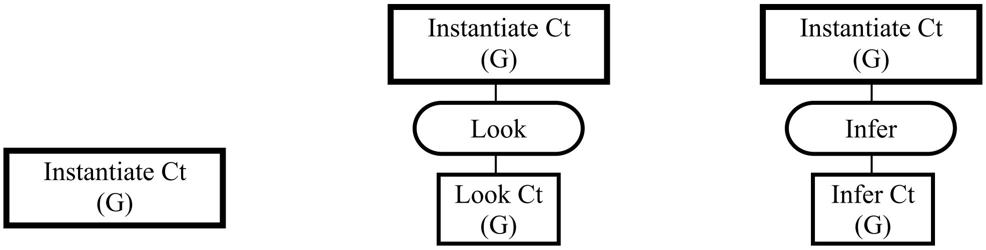
(a) A general recipe for the GIS to instantiate a parameter P with a vague spatial concept VSC with major context information. *Calculate VSC* gives the parameter value based on the system's model of the meaning of the vague spatial concept.
T_Pi: parameter representing a task context factor; *S_Pi*: parameter representing a spatial context factor; *H_Pi*: parameter representing a human context factor



(b) A recipe for the GIS to calculate a distance value from the vague spatial concept *near* with major context information

Goal: user's goal behind the spatial information request;	Vehicle: user's transportation mode;
Scale: spatial scale of the interested area;	U: user name
Refer.: Reference location;	Target: Target location
Routes: Spatial distribution of routes	

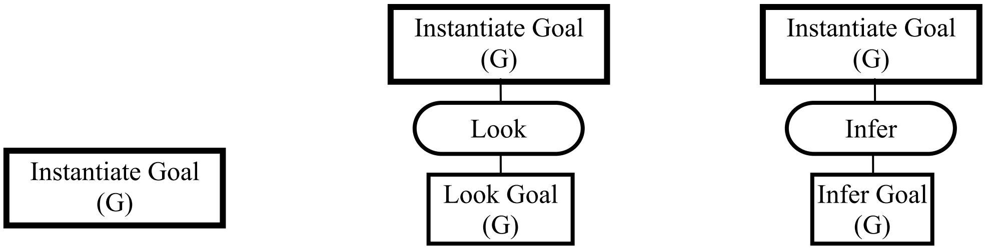
Figure **A-9**: Recipe of *Infer VSC*



(a) Action for the GIS to instantiate a parameter *Ct* representing context information

(b) Recipe for the GIS to look for the parameter value for *Ct* (representing context information) from the existing *PlanGraph*

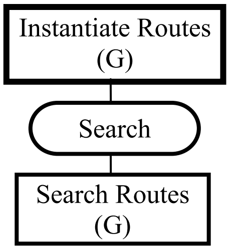
(c) Recipe for the GIS to infer the parameter value for *Ct* representing context information



(d) Action for the GIS to instantiate the parameter *Goal* representing context information

(e) Recipe for the GIS to look for the parameter value for *Goal* (representing context information) from the existing *PlanGraph*

(f) Recipe for the GIS to infer the parameter value for *Goal* representing context information



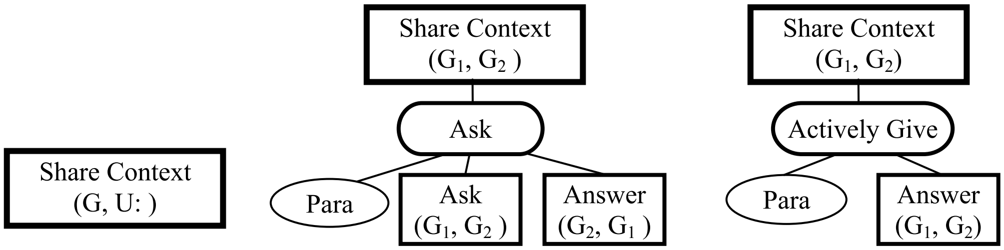
(g) Recipe for the GIS to search map layers representing routes

Figure **A-10**: Actions and Recipes for Inferring Context Information

A.3 Collaborative Human-GIS Dialogues

A.3.1 For Context-Sharing

Here are actions and recipes for the GIS agent and the user to share context information (Figure **A-11**).



(a) Communicative action for the GIS and the user to share context information with each other

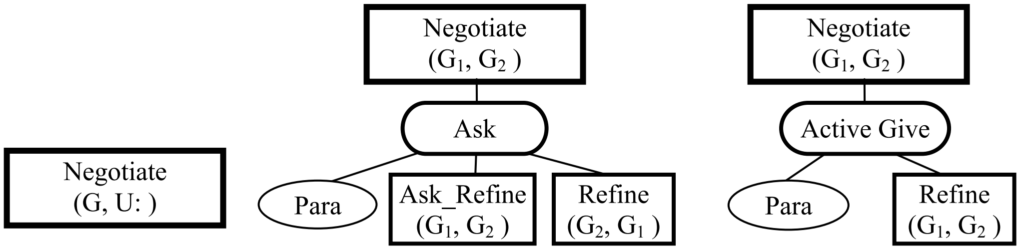
(b) Recipe of *Share Context* for the agent G_1 to ask the other agent G_2 for and wait G_2's reply on the parameter value of *Para*, representing context information

(c) Recipe of *Share Context* for G_1 actively to share the parameter value of *Para*, representing context information with G_2

Figure **A-11**: Actions and Recipes for Context-Sharing

A.3.2 For Negotiation

Here are actions and recipes for the GIS agent and the user to negotiate the meaning of a vague spatial concept (Figure **A-12**).



(a) Communicative action for the GIS and the user to negotiate the meaning of a vague spatial concept

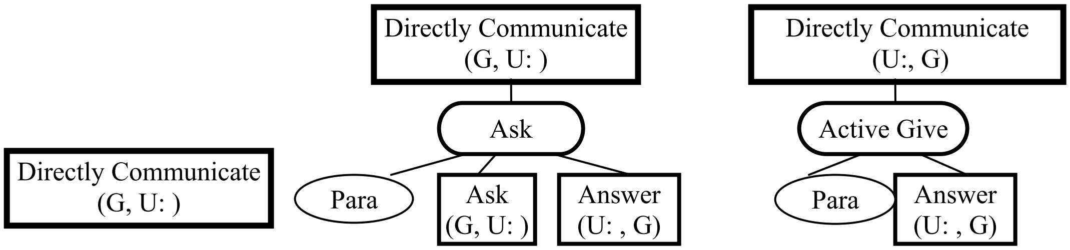
(b) Recipe of *Negotiate* for G_1 to ask for G_2's comment about how to refine the parameter value of *Para*, representing the meaning of a vague spatial concept

(c) Recipe of *Negotiate* for G_1 to actively share with G_2 with his/her commonts on how to refine the parameter value of *Para*, representing the meaning of a vague spatial concept

Figure **A-12**: Actions and Recipes for Negotiation

A.3.3 For Direct Communication

Here are actions and recipes for the GIS agent and the user to directly communicate the meaning of a vague spatial concept (Figure **A-13**).



(a) Communicative action for the GIS and the user to communicate the meaning of the vague spatial concept directly

(b) Recipe of *Directly Communicate* for the GIS to ask the user for and wait for the user's reply on the parameter value of *Para*, representing the meaning of a vague spatial concept

(c) Recipe of *Share Context* for the GIS to understand the user's active providing the parameter value of *Para*, the meaning of a vague spatial concept

Figure **A-13**: Actions and Recipes for Direct Communication

Appendix B

Mental State Numbers in *PlanGraph* Implemented in *GeoDialogue*

B.1 MSNs on Action Nodes in *PlanGraph*

Table **B-1**: Mental State Numbers (MSNs) on Action α and Their Meanings

Label		Intention on doing α	Commitment on success of α	Belief (the agent being modeled believes that)
0		individual	individual	α will be part of collaboration.
1		Individual & group	Individual & mutual	α is part of collaboration. The other agent in collaboration also believes that α is part of collaboration.
2		Individual & group	Individual & mutual	α is basic; The other agent in collaboration would agree to execute it directly.
3		Individual & group	Individual & mutual	α is complex; The other agent in collaboration would agree the particular recipe selected for α.
4	4_1	individual & group	Individual & mutual	The current plan for α does not succeed, and α can be re-executed with another plan.
	4_2a	individual & group	Individual & mutual	The current plan for α to infer the meaning of a vague spatial concept may not succeed because the context information used in this plan is not shared between the agent being modeled and the other agent(s) in collaboration.
	4_2b	individual & group	Individual & mutual	The current plan for α to infer the meaning of a vague spatial concept does not succeed because the context information used in this plan is not shared between the agent being modeled and the other agent(s) in collaboration.
	4_3b	individual & group	Individual & mutual	The current plan for α may not succeed although the context information used in this plan has been shared between the agent being modeled and the other agent(s) in collaboration.
	4_4a	individual & group	Individual & mutual	The current plan for α does not succeed because its descendants involves a parameter instantiated from a vague spatial concept.
	4_4b	individual & group	Individual & mutual	The current plan for α may not succeed because its descendants involves a parameter instantiated from a vague spatial concept.
5		no	no	α fails and there are no other plans available to re-perform it or there is no way to improve α.
6		individual & group	Individual & mutual	α was successfully finished.

Note: In this table, “the agent being modeled” means the agent whose dynamic knowledge is represented with the *PlanGraph* model. In this study, the agent being modeled is the GIS agent.

B.2 MSNs on Parameter Nodes in *PlanGraph*

Table **B-2**: Mental State Numbers (MSNs) on Parameter *p* and Their Meanings

Label		Intention	Commitment	Belief (the agent being modeled believes that)
0		individual	individual	Instantiation of p is part of collaboration;
1		individual & group	Individual & mutual	The user also believes that instantiation of p is part of collaboration.
2		individual & group	Individual & mutual	p is optional and is being instantiated with a plan.
3		individual & group	Individual & mutual	p is required and is being instantiated with a plan.
4	4_1	individual & group	Individual & mutual	The current plan for instantiation of p does not succeed, and p can be re-instantiated with another plan.
	4_2b	individual & group	Individual & mutual	The value of p may need to be further improved and certain context information used for instantiation of p is not shared between the agent being modeled and the other agent(s) in collaboration
	4_2a	individual & group	Individual & mutual	The value of p needs to be further improved and certain context information involved in instantiation of p is not shared between the agent being modeled and the other agent(s) in collaboration.
	4_3b	individual & group	Individual & mutual	The value of p may need to be further improved and context information involved in instantiation of p has been shared between the agent being modeled and the other agent(s) in collaboration
	4_3a	individual & group	Individual & mutual	The value of p needs to be further and context information involved in instantiation of p has been shared between the agent being modeled and the other agent(s) in collaboration
	4_4b	individual & group	Individual & mutual	The value of p whose descendants involves a parameter instantiated from a vague spatial concept may need to be improved.
	4_4a	individual & group	Individual & mutual	The value of p whose descendants involves a parameter instantiated from a vague spatial concept needs to be improved.
5		no	no	Instantiation of p fails.
6	6_1	individual & group	Individual & mutual	Instantiation of p was successfully finished for sure.
	6_2	individual & group	Individual & mutual	Instantiation of p was successfully finished with estimation.
	6_3	individual & group	Individual & mutual	Instantiation of p was successfully finished, but the agent being modeled does not have such knowledge about *p*'s value.

Note: In this table, "the agent being modeled" means the agent whose dynamic knowledge is represented with the *PlanGraph* model. In this study, the agent being modeled is the GIS agent.

Appendix C

Reasoning Algorithms in *PlanGraph* in *GeoDialogue*

C.1 Interpretation Algorithms

The following **IA**s are designed and implemented for the case that the *PlanGraph* is empty and the user's request initiates the human-GIS communication:

- **IA1**: If the GIS thinks that the success of the user's intention (represented as an action α) underlying the user's input needs the GIS agent's participation (the GIS knows that it can help the user's intention), the GIS initiates the *PlanGraph* with α as the root action, and places the attention on α.
- **IA2**: If the system thinks that the user's intention represented as α underlying the input may involve spatial information requests which the GIS can help the user with, the GIS initiates the *PlanGraph* with α as the root action, and places the attention on α. This algorithm is designed for the GIS to take the user's goal as part of the context behind human-GIS communication in case that the user's goal does not directly involve the GIS's participation, but indirectly involve spatial information that the GIS can help the user with.

The following IAs are designed and implemented for this case that the existing *PlanGraph* is not empty and the current focus in *PlanGraph* is β:

- **IA3**: If α is an instantiation of β or instantiation of a sub action γ under β, the system needs to replace β or γ with α, and places its attention on β.

- **IA4**: If α is part of a recipe R for β and constraints associated with R are met, the system needs to replace β with the recipe R in the *PlanGraph*, updates MSN on β as 3, and place its attention on β or β's first descendant or β's first ascedent which the GIS agent can may participate in.
- **IA5**: If the focus action β is part of α after elaboration of α with certain recipes Rs, and constraints required for Rs are met, the system needs to integrate Rs for α with the *PlanGraph*, updates MSN on α as 3, and place attention on β after elaborating α with information available. This algorithm is designed for the GIS to understand the user's goal under the current utterance as the context involving the current focus on the *PlanGraph*.
- **IA6**: If α is part of the focus action β after elaboration of β with certain recipes Rs and constraints required for Rs are met, the system needs to integrate α with the *PlanGraph* through Rs, updates MSN on α as 3, and place attention on α after elaborating β with information available. This algorithm is designed for the GIS to integrate the user's spatial information request with the context as the current focus on the *PlanGraph*.
- **IA7**: This algorithm works in the case that the user's intention represented as the action α is to improve the current parameter value of a parameter *p* around the focus action β in the *PlanGraph*:
    - (a): α is to instantiate a parameter *p* in β in *PlanGraph,* the GIS moves attention to *p* after updating its MSN on β as 3 and MSNs on subactions in β as 1.

  - (b): α is to instantiate a parameter *p* in one of descendants of β in *PlanGraph,* the GIS updates its MSN on β as 3 and MSNs all descendant actions which need to be executed after instantiation of *p* as 1.
  - (c): to instantiate a parameter *p* in the focus action β's ascendant action, and the system's MSN on the parameter *p* is not 6_1 (that is, p's existing *value* is not instantiated for sure), then, the system moves attention back to β after completing instantiating the parameter *p* (updates its MSN on *p* and attaches α to the parameter *p* later).

- **IA8**: If the user's intention represented as the action α is to express the user's mental attitude (agreement or disagreement) on the focus action β, and the focus action β in *PlanGraph* has MSN as 4_4b, the GIS updates MSN on β in terms of the user's attitude.This algorithm is designed for the GIS to recognize the user's mental attitude on the result of human-GIS collaboration.
- **IA9**: If the user's intention represented as the action α is to express the user's new information request (different from the original one involving the vague spatial concept), and the focus action β in *PlanGraph* representing the last spatial information request has MSN as 4_4a, the GIS agent changes MSN on β as 4_4b and integrate the new information request with the existing *PlanGraph* after checking the parent action γ of β. This algorithm is specifically designed for the GIS to recognize the user's new spatial information request when the last spatial information still needs to be improved.

- (a): if the required information corresponds to one of parameters involved in the existing plan of γ, the GIS agent interprets α as initiating *Share Context* with the recipe *Ask* and moves attention back to β after finishing *Ask*.
- (b): if the existing plan for γ does not include a parameter corresponding to the new required information, the GIS agent can re-elaborate γ so that the parameter corresponding to the new information requested is included in the new plan for γ, the GIS agent interprets α as initiating *Share Context* with the recipe *Ask* and moves attention back to β after finishing *Ask*;
- (c): if the existing plan for γ does not include a parameter corresponding to the new required information, the GIS agent fails to include the parameter corresponding to the new information requested in a new plan for γ, the GIS agent directly interprets the new information request and moves attention back to β after finishing elaborating the new request.

C.2 Output Algorithms

Two **ORs** are described as follows.

OR1: If the focus action α is a communicative action to be initiated by the GIS, the system outputs the responses that have been generated so far (including map responses and the system's explanation/questions), updates MSN on α as 6 and moves attention to next sibling node if any or the parent of α.

OR2: If the focus action is the last action that the GIS can participate in *PlanGraph*, the system outputs the responses. This algorithm is designed by following the theorem in the *Joint Intention* theory (Cohen & Levesque, 1991) that, as to a jointed commitment, the individual agent should keep other agents in collaboration mutually known the status of the joint commitment.

C.3 Elaboration Algorithms

C.3.1 Elaboration Algorithms for Actions

In case that the focus action α is non-communicative, a set of EAAs are designed and implemented for the GIS agent to reason what, when and how to do next with α in terms of the GIS agent's MSN on α:

- **EAA0**: In case that the MSN on α is 0, the GIS directly changes MSN as 1. In this study, the GIS keeps default MSNs as 0 on all actions that are just added in *PlanGraph*. With this algorithm, the GIS agent changes the initial intention, commitment and belief on α from individual to joint/mutual. This algorithm is designed under the assumption that the user always believes the GIS agent's capability to help the user for the user's spatial information request and the GIS always knows that the user believes its capabilities unless that the GIS explicitly informs that it can not fulfill the user's request.
- **EAA1**: In case that the MSN on α is 1:
    - If α is basic, the system updates its MSN on it as 2;

  - If α is complex, the system finds a suitable recipe from its knowledge base if possible, integrates this recipe to the *PlanGraph* and updates its MSN on α as 3;
  - If α is complex, and there is no suitable recipe (the system does not know how to do α), currently, the system updates its MSN on it as 5 and generate a response to the user which indicates the failure on this action;
- **EAA2**: In case that the MSN on α is 2, the system directly executes it and updates its mental status depending on the execution result.
- **EAA3**: If the MSN on α is 3,
  - a)if there are any parameters/subactions under α that have not been finished, the system places the attention on the first parameter/subaction under α to be handled.
  - b) in case that α is not to instantiate a parameter value from a vague spatial concept, and all its subactions have been finished, the GIS agent update its MSN on α in terms of the results of instantiation of all parameters and execution of all subactions involved in the plan for α. For example,
    - 1) if all its α's parameters and sub actions have been finished successfully, the system updates MSN on α as success (6);
    - 2) if part of its subactions/parameters has MSN as 4, the system updates MSN on α as 4_4b.

    - 3) if one of child nodes has MSN as 5, the system changes its MSN on α as 4_1 if there is another appropriate recipe for α, otherwise keeps its MSN on α as 5 if no more appropriate recipe for α exists.
  - c) in case that α is to instantiate a parameter value from a vague spatial concept, and all its subactions have been finished, the GIS agent update its MSN on α in terms of the results of instantiation of all parameters and execution of all subactions involved in the plan for α:
    - if all the parameters and subactions under α have been finished successfully for sure (all MSN on subactions are 6 and all MSN on parameters are 6_1 or 6_3), the system updates MSN on α as 4_3b. In this case, context information used in the plan to infer the meaning of the vague spatial concept has been shared between the GIS and the user.
    - if part of parameters (representing context information) under α are estimated by the GIS (MSNs on them are 6_2), the system updates MSN on α as 4_2b. In this case, certain context information used in the plan to infer the meaning of the vague spatial concept has not been shared between the GIS and the user.
- **EAA4**: Here gives details of a few example **EAA4**:
  - _1) If MSN on α = 4_1, the system finds another recipe R for α (or another way to do α), replace α with R in *PlanGraph*, updates the mental status on α as 3, and places its attention on α . This algorithm is designed for the GIS to reason how to elaborate a regular action α in case that the

plan for α fails and another appropriate recipe for α exists. This algorithm follows the way by which the human communicator tries different plans to do an action if the current plan for this action fails.

- _2)
    - (b) if MSN on α = 4_2b, the system moves attention to α’s parent node, that is, the parameter instantiated from the vague spatial concept. This algorithm is designed for the GIS to continue the current plan for α toward returning the user a response with an inferred meaning of the vague spatial concept.
    - (a) if MSN on α = 4_2a, the system changes MSN on α as 3, and changes MSNs on α’s child nodes to improve the existing plan for α (in case that MSN on α’s parameter is 6_2, the system changes its MSN on the parameter as 1; in case that MSN on α’s sub action is 6, the system changes its MSN on the sub action as 1). This algorithm is designed for the GIS to improve the previous plan to infer the meaning of the vague spatial concept by improving instantiation of optional parameters representing major context information involved in this previous plan. Improving instantiation of these optional parameters with estimated values will lead to collaborative human-GIS dialogues for *Share Context*.
- _3) if MSN on α = 4_3b, the system moves attention to α’s parent node, that is, the parameter instantiated from the vague spatial concept and updates its MSN on this parameter as 4_3b. This algorithm is designed for

the GIS to continue the current plan for α towards giving the user a response with an inferred meaning of the vague spatial concept.

- _4)
  - if MSN on α = 4_4b, the system generates a response indicating the result of execution of α with an inferred meaning of the vague spatial concept if α directly involves a parameter instantiated from a vague spatial concept, and moves attention to α's next sibling node if any, or moves attention to α's parent node that the GIS can participate in if no sibling action node exists, or if α is the last action in *PlanGraph* that involves the GIS agent's participation, the GIS keeps attention on α after generating a response indicating the vagueness problem (e. g. "Is this OK?"). This algorithm is designed for the GIS to continue the current plan to execute a GIS command with an inferred meaning of the vague spatial concept.
  - if MSN on α = 4_4a, the system changes MSN on α as 3, and changes MSNs on α's child nodes:
    - If the user's newest utterance contains information about revising any parameter *p* of α, the GIS agent changes MSN on *p* as 1 and MSN on all subactions under α as 1, moves attention to only *p* and all subactions under α; otherwise
    - The GIS agent changes MSNs on α's child nodes (in case that MSN on α's parameter is 4_2b, 4_3b or 4_4b, the system changes its MSN on the parameter as 4_2a, 4_3a ir 4_4a correspondingly;

in case that MSN on α's sub action is 6, the system changes its MSN on the sub action as 1), and keeps attention on α. This algorithm is designed for the GIS to come back to re-execute the GIS command with a better understanding about the vague spatial concept if the user does not agree with its previous understanding about the vague spatial concept.

- **EAA5**: If MSN on α is 5, the system generates a response indicating failure on execution of α for later output, and moves attention on α's parent node that the GIS can participate in if any or keeps attention on α if α is the last action in *PlanGraph* that the GIS can participate in. This algorithm is designed for the GIS to communicate its belief and commitment with the user.
- **EAA6**: In case that MSN on the action α is 6:
  - if α is not the last action in *PlanGraph* that the GIS can participate in, the system moves its attention to α's next sibling node to be done if any, or moves its attention on α's parent node if α's parent node involves the GIS's participation;
  - if α is the last action in *PlanGraph* that the GIS can participate in (in this study, usually the last action involving the GIS's participation is the action representing the user's spatial information request), keeps attention on α and waits for the user's next request. This algorithm is designed for the GIS to focus on actions that the GIS can participate and not to elaborate the actions that do not involve the GIS's participation, e. g. the user's final goal behind the user's map request.

In case that the focus action α is communicative, the following are a set of EAAs for the GIS agent to reason what, when and how to do next with α in terms of the GIS agent's MSN on α:

- **EAAU**: In case that the system's current attention is on a non-communicative action α to be done by the user,
    - If α is not the parent of the action representing the user's spatial information request, then the GIS changes its MSN on the action α as 6. This reasoning algorithm is designed under the assumption that the GIS agent believes that the user can successfully finish the user's duty in the human-GIS collaboration.
    - If α is the parent of the action β representing the user's spatial information request, the GIS moves attention to the next action on *PlanGraph* that the GIS can participate in.
- **EAAC**: In case that the system's current attention is on a communicative action α, if α is to be initiated by the system, the GIS changes its MSN on α as 6 after generating the response to the user under the assumption that the GIS usually can successfully send out the response to the user; if α is to be initiated by the user, the GIS changes its MSN on α in terms of the result of interpretation of the user's input (if the user's input satisfies the expected need on α, then the GIS changes its MSN on α as 6, otherwise as 5).

C.3.2 Elaboration Algorithms for Parameters

A set of **EAP**s are designed as follows:

- **EAP0**: In case that the MSN on *p* is 0, the GIS directly changes MSN as 1. In this study, the GIS keeps default MSNs as 0 on all parameters in actions/their recipes that are integrated in *PlanGraph*. With this algorithm, the GIS agent changes its intention and commitments on *p*'s instantiation from individual one to joint one. This algorithm is designed under the assumption that the user and the GIS in collaboration always commit to helping each other on instantiating parameters involved in the collaboration.
- **EAP1**: In case that the MSN on *p* is 1, the GIS agent attaches an action α with *p* to instantiate *p*, places attention on α and updates MSN on *p* as 2 or 3 depending on that if *p* is optional or required. In case that *p* is an optional parameter representing a context factor involved in an action to instantiate a parameter from a vague spatial concept:
    - a) If *p* does not have an estimated parameter value already (which means that this is the first time for *p* to be elaborated), the GIS agent needs to check if the corresponding part *p'* (corresponding to *p*) in the action β representing the context directly behind the user's spatial information request to the GIS exists in the existing *PlanGraph* first before moving attention on further elaboration of the action *Instantiate Ct* to instantiate *p*:
        - If *p'* exists in β, the GIS agent moves attention on further elaboration of *Instantiate Ct;*

    - If not, before moving attention back to *Instantiate Ct*, the GIS agent needs to elaborate β until *p'* can be found in the existing *PlanGraph*. During the process of elaboration of β, the GIS agent needs to elaborate parameter nodes and ignore the action nodes to be done by the user. If a parameter *q* in β represents spatial information needed in β, which is not the spatial information requested by the user, the GIS agent can actively show the parameter value of *q* if it is not currently on the map in order to build a shared context with the user. In case that *p* represents the user's goal in the context behind the user request and β does not exist in the existing *PlanGraph*, the GIS agent needs to instantiate *p* first to determine β, and then integrate β with the existing *PlanGraph* before instantiating other optional parameters representing other context factors. In this study, the GIS can either infer it or directly use the communicative action *Share Context* to determine it.
- b) If *p* has an estimated parameter value *pv* already (which means that the GIS agent re-elaborates *p*), the GIS agent needs to check if *p'* in β first before moving attention on further elaboration of the action *Instantiate Ct* to instantiate *p*:
    - If *p'* exists in β and *p'* has a new value different from *pv*, the GIS agent moves attention on further elaboration of *Instantiate Ct;* If *p'* does exist in β and *p'* has a value same as *pv*, the GIS agent changes MSN on *p'* as 0, and attaches an communicative action *Share Context* with *p*'. If *p*' is the human agent in the plans representing the human-GIS

    collaboration, the GIS agent directly attaches *Share Context* with *p* and updates all *p'* after instantiation of *p*. This algorithm is designed by following the HCP2(c).

    - If *p'* does not exists in β, before moving attention back to *Instantiate Ct*, the GIS agent needs to elaborate β until *p'* can be found in the existing *PlanGraph*. During the process of elaboration of β, the GIS agent needs to elaborate parameter nodes and ignore the action nodes to be done by the user. If *p* represents the user's goal in the context behind the spatial information request, the GIS agent needs to determine β first.

- **EAP2/3**: In case that the MSN on *p* is 2/3, the system updates its MSN on *p* depending on the result of the action α under *p* to instantiate *p.* For example:
    - a) if MSN on α is 4_2b, 4_3b, 4_4b (this means that *p* is instantiated from a vague concept or instantiated from a GIS command involving a parameter instantiated from a vague spatial concept, and may need improvement later), the system updates MSN on *p* as 4_2b, 4_3b, or 4_4b correspondingly; An exceptional case is that, if *p* is re-estimated with a same value as before, the system directly updates MSN on *p* as 4_2a, 4_3a, or 4_4a correspondingly;
    - b) if MSN on α is 6 and α is a communicative action for the GIS to achieve the parameter value for *p* which was instantiated from a vague spatial concept, the system updates MSN on *p* as 4_3b;

  - c) if MSN on α is 6 and *p* is not a parameter which was instantiated from a vague spatial concept, the system updates its MSN on *p* as 6_1 (if *p* is instantiated with information from the human-GIS dialogue or the GIS's spatial database) or 6_2 (if *p* is instantiated from a recipe to infer the parameter value).
  - d) if MSN on α is 5, the system updates MSN on *p* as 4_1 if there is another appropriate way (another recipe for α or use communicative actions) to instantiate *p*; otherwise the system updates its MSN on *p* as 5.
- **EAP4:** In case that the MSN on *p* is 4, the GIS either continue to elaborate the remaining parameters/subactions in *PlanGraph* with an inferred parameter value for *p* from a vague spatial concept towards returning spatial information requested by the user, or try to improve the parameter value of *p* after receiving the user's negative comments on the existing parameter value for *p.*
  - _1) if MSN on *p* is 4_1, the system updates its MSN on *p* as 2 or 3 depending on whether *p* is optional or required, and moves attention to its child action.
  - _2)
    - if MSN on *p* = 4_2b, the system generates a response indicating inferred context information used to infer the meaning of the vague spatial concept, and moves attention to *p*'s sibling node. This algorithm is designed for the GIS to continue to execute the GIS command with an inferred understanding of the vague spatial concept with inferred context information and inform the user the inferred context

information later by returning a speech response together with the map response.

    - if MSN on *p* = 4_2a, the system changes MSN on *p* as 2 or 3 depending on whether *p* is optional or required, and places attention on improving instantiation of *p* with the previous plan or the new plan suggested by the user. In the former case, the system moves attention to the previous plan R under *p*, and changes MSN on R as 4_2a.
- _3)
    - if MSN on *p* = 4_3b, the system moves attention to *p*'s sibling node.
    - if MSN on *p* = 4_3a, the system changes its MSNs on *p* and its child action as 4_1.
- _4)
    - if MSN on *p* = 4_4b, the system moves attention to *p*'s next sibling node. This algorithm is designed for the GIS to continue to execute certain actions with an inferred understanding of the vague spatial concept.
    - if MSN on *p* = 4_4a, the system changes MSN on *p* as 2 or 3 depending on whether *p* is optional or required, and places attention on improving the action to instantiate *p* with the new plan proposed by the user or the previous plan. In the later case, the system moves attention to the previous plan R under *p*, and changes MSN on R as 4_4a. This algorithm is designed for the GIS to re-visit the previous plan to

instantiate the parameter value in case that the GIS and the user have not reached agreement on the existing parameter value of *p*.

- **EAP5**: If MSN on *p* = 5, the system generates a response indicating the failure of *p's* in instantiation for later output and moves attention on *p*'s parent node in *PlanGraph*. This algorithm is designed for the GIS to communicate its belief (failure) on instantiation of *p* to the user.
- **EAP6**: In case that MSN on *p* is 6 (6_1, 6_2, or 6_3), the GIS moves attention to the next node in the *PlanGraph* that the GIS needs to elaborate.
  - If *p* represents the user's goal in the context behind the user request and the GIS agent keeps MSN on p as 6_2, the GIS agent needs to integrate the action (MSN 3) representing the user's goal (with an appropriate recipe) with the existing *PlanGraph* before moving attention to the next node in the *PlanGraph*.

VITA

Hongmei Wang received the B. E. degree from Department of Geoscience, Changchun College of Geology, P. R. China, in 1994, and the M. S. degree from Institute of Remote Sensing Applications, Chinese Academy of Sciences, P. R. China, in 1997. She had worked in Institute of Remote Sensing Applications, Chinese Academy of Sciences, P. R. China, as a Research Associate from August 1997 to August 2000. From September 2000 to July 2001, she had been a Master student in Department of Geography, York University, Canada. Since August 2001, she has been a Ph. D student in the College of Information Sciences and Technology, The Pennsylvania State University, University Park, PA. Her current research interests lie on the boarder of computer science and geography.