

Implementação de um Tradutor de Linguagem Natural para um Sistema Geográfico*

Sérgio M. B. Lima, Ariadne M. B. R. Carvalho e Claudia B. Medeiros

IC - UNICAMP - CP 6176

13083-970 Campinas-SP-Brazil

Fax: 55 (019) 788-7470

{sergiomu,ariadne,cmbm}@dcc.unicamp.br

Resumo

Este trabalho descreve a implementação de um tradutor de língua natural para a linguagem de consultas espaciais LEGAL, combinando pesquisas em língua natural e bancos de dados espaciais. A motivação para o trabalho surgiu da necessidade de auxiliar os usuários de Sistemas de Informação Geográfica, que normalmente não são especialistas em computação, a formular consultas através de língua natural. As vantagens e desvantagens desse tipo de interação são apresentadas, bem como as funcionalidades de cada módulo do sistema, os problemas lingüísticos enfrentados durante a implementação do tradutor e as particularidades das consultas espaciais tratadas pelo sistema.

Palavras-chaves: Bancos de Dados, Inteligência Artificial, Sistemas Geográficos

1 Introdução

Sistemas de Informação Geográfica – SIGS – são sistemas automatizados usados para armazenar, analisar e manipular dados geográficos. A heterogeneidade das aplicações e tipos de dados manipulados por tais sistemas dificulta a tarefa do usuário, que além de ser especialista em um determinado domínio de aplicação (gerenciamento ambiental, biologia, geologia, planejamento urbano, por exemplo), precisa também conhecer diferentes linguagens de acesso a cada SIG específico.

Todos os SIGS modernos têm como base algum Sistema Gerenciador de Bancos de Dados (SGBD), que armazena tanto os dados convencionais quanto sua descrição espacial (os chamados *dados geo-referenciados*). O termo denota dados que descrevem fatos, objetos e fenômenos do globo terrestre associados à sua localização sobre a superfície terrestre, num certo instante ou período de tempo.

Consultas a dados geo-referenciados podem ser feitas segundo dois tipos de paradigmas – textual e visual. O primeiro, mais comum, é baseado no uso de linguagens de consulta que estendem o padrão SQL de bancos de dados relacionais com operadores espaciais. O segundo envolve uso de ícones e grafos para formulação de consultas, dando margem a vários problemas de interpretação semântica. Sistemas comerciais permitem, igualmente, que consultas simples possam ser realizadas através de interação direta (usando o *mouse*), mas consultas espaciais complexas exigem programação em linguagem textual, para acesso ao SGBD subjacente.

Existem várias propostas para diminuir o *gap* entre o usuário e o SIG, através do aperfeiçoamento da interface. Linguagens de consulta a bancos de dados em geral se mostram inadequadas, pois para utilizá-las o usuário necessita saber detalhes sobre o banco de dados (modelo de dados, esquema, restrições), além de se deparar com sintaxes artificiais e às vezes de difícil aprendizado [Wang94]. A solução adotada neste artigo envolve o uso de língua natural (LN).

*Trabalho com apoio financeiro da CAPES, CNPq, FAPESP e projetos CNPq PROTEM/CC-GEOTEC, Comunidade Européia – projeto ICDT 116.

A motivação para nosso trabalho surgiu da necessidade da construção de uma interface adequada à formulação de consultas por usuários de um SIG. Um aspecto original do trabalho é o fato de combinar resultados em LN, bancos de dados espaciais e SIGs. Como primeiro resultado, descrevemos a implementação de um protótipo experimental de um tradutor de LN para a linguagem LEGAL (Linguagem Espacial para Geoprocessamento Algebrico) [Cam95], desenvolvida no INPE para o SIG SPRING. As principais contribuições apresentadas são:

- Do ponto de vista de LN, estudo dos fenômenos lingüísticos particulares à manipulação de dados espaciais armazenados em SIGs;
- Do ponto de vista de bancos de dados, análise das dificuldades de fornecer dados para permitir a condução de consultas espaciais formuladas em LN.

O restante deste artigo está organizado da seguinte forma. A seção 2 fornece uma descrição sucinta de LEGAL, para permitir o entendimento do texto, bem como apresenta um pequeno banco de dados geográfico que será utilizado para ilustrar os exemplos do artigo. A seção 3 detalha a arquitetura do protótipo implementado, e mostra como ele pode processar consultas sobre o banco de dados da seção 2.2. A seção 4 analisa os problemas lingüísticos encontrados, e a seção 5 os problemas de processamento de consultas espaciais formuladas em LN do ponto de vista de bancos de dados. A seção 6 fornece uma visão geral de trabalhos correlatos e a seção 7 conclui o artigo.

2 A linguagem LEGAL

Um SIG permite manipular dois tipos básicos de dados: geo-referenciados (de agora em diante também chamados *dados espaciais*) e convencionais (ou não espaciais). Uma característica particular dos dados geo-referenciados é a quantidade de *representações* diferentes de uma mesma entidade do mundo real, conforme varie a escala em que é armazenada, a projeção cartográfica escolhida, a época de aquisição do dado, ou mesmo conforme a visão do usuário ou aplicação.

2.1 Características básicas da linguagem

LEGAL é uma linguagem de consulta e manipulação espacial utilizada pelo SIG SPRING, desenvolvido no INPE. Ela estende o SQL com operadores espaciais e facilidades para apresentação de dados espaciais. Além de consultas propriamente ditas, LEGAL permite também definir bancos de dados espaciais. Este trabalho se restringe às características de consulta, ignorando as de definição de bancos de dados e de apresentação. Aqui mostramos apenas os detalhes da linguagem necessários ao entendimento do texto. Para uma descrição detalhada, remetemos o leitor às páginas [www](#) [WWW96].

Uma consulta em LEGAL possui dois componentes: uma expressão de busca representada em SQL estendido e uma resposta à consulta, que pode ser objeto de manipulação posterior. A linguagem permite manipular vários tipos de representações, incluindo imagens e modelos numéricos de terreno. Consultas complexas podem ser expressas sob forma de programas, com declaração de variáveis e chamadas de operações espaciais sobre as diferentes representações. Cada tipo de representação permite operações diferentes, de forma similar à orientação a objetos. Por exemplo, operações realizadas sobre imagens de entidades espaciais não são as mesmas que aquelas aplicadas sobre representações em que estas entidades são representadas como polígonos.

Por exemplo, considere as entidades espaciais "Município" e "Cidade". A consulta "*Selezione os nomes de todos os municípios a menos de 50 km da cidade de Belo Horizonte*", retirada de [CCH⁺96], é formulada em LEGAL através do comando:

```
SELECT m.nome
FROM m Municipio, c Cidade
WHERE c.nome = 'Belo Horizonte' AND distance(m,c) < 50
```

Um grande problema no processamento de consultas em SIGs reside exatamente na utilização dos operadores espaciais adequados a cada representação. Por exemplo, o cálculo da distância entre duas cidades representadas por pontos em uma certa escala é realizado de forma diferente se estas estão armazenadas em uma escala maior e representadas por regiões. Assim, o processamento de consultas neste contexto é dependente da representação com que se trabalha. Quando um objeto espacial tem mais de uma representação, LEGAL oferece a opção de escolha, usando a cláusula **REPRESENTED BY**. Por exemplo, se houver mais de uma representação para Municípios e Cidades, a consulta acima pode ainda ser formulada em LEGAL escolhendo a representação adequada dentro de um conjunto *Mapa*

```
SELECT m.nome
FROM m Municipio, c Cidade
WHERE c.nome = 'Belo Horizonte' AND distance(m,c) < 50
      REPRESENTED BY (SELECT rep
                      FROM rep Mapa
                      WHERE rep.escala = '1:100.000')
```

A cláusula **REPRESENTED BY** indica que as localizações dos objetos em “Município” e “Cidade” deverão ser obtidas a partir da representação de escala 1:100.000. Note que para outra escala (por exemplo, 1:10.000) o cálculo de distância não pode ser implementado da mesma forma. A designação de representação pode ser omitida caso haja apenas uma representação para um dado objeto espacial.

2.2 Exemplo utilizado

Para ilustrar o texto, utilizaremos um exemplo simples de um banco de dados espacial, onde os objetos espaciais são armazenados como tuplas de relações. A informação espacial propriamente dita é armazenada em um atributo denominado “geometria”, que é formado por listas de coordenadas. O esquema desse banco de dados é:

Cidade(nome, área, população, altitude, prefeito, geometria)
Município(nome, área, população, geometria)
Estado(nome, área, população, governador, geometria)
País(nome, área, população, presidente, geometria)
Rio(nome, extensão, geometria)
Estrada(nome, extensão, geometria)
Lago(nome, área, geometria)
Administrador(nome, sexo, partido).

Os atributos “nome”, “prefeito”, “governador”, “presidente”, “sexo” e “partido” são do tipo texto, enquanto “área”, “população”, “altitude” e “extensão” são atributos numéricos; “geometria” é uma lista de coordenadas e todas as relações, exceto “Administrador”, referem-se a entidades geo-referenciadas.

3 A Arquitetura do Protótipo

O sistema foi implementado em PROLOG [CM87] e é composto por cinco módulos: o analisador, o léxico (ou dicionário), a base de conhecimentos, o interpretador e o gerador de consultas em LEGAL (veja figura 1).

3.1 O Analisador

O analisador tem a função de analisar sintaticamente a consulta em LN e extrair o seu significado. Para isso, usa uma gramática de cláusulas definidas (GCD) [PW80] capaz de tratar um sub-domínio da língua portuguesa; especificamente, esta gramática tem poder de expressão em relação ao domínio da aplicação.

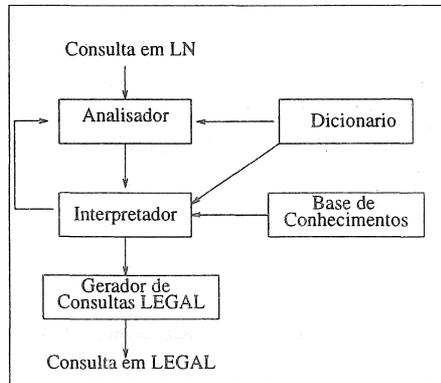


Figura 1: Arquitetura do protótipo implementado

As GCD tem como vantagem a modularidade e a simplicidade na descrição das regras gramaticais, além do tratamento da recursividade, que é uma característica da maioria das línguas de interesse.

Quando a consulta está de acordo com as regras gramaticais, os seus itens (relações, atributos e condições) são extraídos e a consulta em LEGAL é gerada. Quando a análise da sentença falha, o controle é passado para o interpretador.

3.2 O Interpretador

O interpretador permite corrigir tres tipos de problemas: (a) erro ortográfico; (b) palavra que não consta do dicionário; ou (c) erro gramatical. Quando um erro ortográfico é encontrado, a análise é refeita automaticamente com a palavra correta sendo utilizada (por isso existe uma seta retornando do módulo interpretador para o analisador na figura).

Algumas vezes, a consulta pode estar gramaticalmente correta mas alguma relação e/ou atributo pode não estar contido no banco de dados. O desconhecimento do domínio pode levar o usuário a formular consultas sobre objetos que não estejam no banco dados. Quando isso ocorre, o interpretador avisa o usuário que tais objetos não fazem parte do banco de dados. Para atenuar esses erros o usuário pode utilizar as chamadas meta-consultas, buscando por informações contidas na base de conhecimentos. Quando o erro é gramatical, o usuário é avisado que deve reformular a sua consulta.

O interpretador é encarregado de outras funções muito importantes. São elas: (a) resolução de ambigüidades (que será discutida na seção 4.2); (b) construção de relacionamentos entre objetos espaciais e não espaciais, quando necessário; (c) conversão de palavras que expressam relacionamentos topológicos para os operadores topológicos da linguagem LEGAL; e (d) gerenciamento de informações relativas ao contexto.

3.3 O Dicionário

O dicionário do sistema contém o vocabulário necessário para processar as consultas em LN. Como se trata de um protótipo, o tamanho do dicionário é limitado. É composto de: palavras da língua portuguesa em geral (como por exemplo artigos, preposições, verbos); palavras relacionadas ao domínio de aplicação (refletindo o esquema do banco de dados, especificando relações, atributos e domínios) e conceitos associados ao domínio da aplicação (por exemplo, termos designando relacionamentos espaciais), além de alguns sinônimos dessas palavras. Alguns exemplos das classes de palavras contidas no dicionário são:

```

eh_substantivo(estrada)
eh_pronome(qual)
eh_verbo(atraversada,cross)
  
```

```
eh_sinonimo(rodovia,estrada)
eh_adjetivo(maior,>).
```

Um quarto tipo de vocábulos do dicionário é reservado para a conversão de palavras em LN para termos em LEGAL. Por exemplo, na consulta “Qual é a população média das cidades com área maior que 100?”, a palavra “média” é convertida para “avg” e a palavra “maior” para o símbolo “>”. A sentença é convertida para os seguintes termos em LEGAL:

```
SELECT AVG(populacao))
FROM c IN cidade
WHERE area > 100
```

3.4 A Base de Conhecimentos

A base de conhecimentos é composta por um conjunto de comandos em PROLOG que descreve o esquema do banco de dados espacial (atributos, relações), restrições de integridade (chaves) e os relacionamentos entre as relações. No banco de dados usado como exemplo estão contidos os nomes de relação relativos a objetos geo-referenciados (“Cidade”, “Estado”, “País”, “Município”, “Rio”, “Estrada”, “Lago”) e da entidade não espacial “Administrador”. Além disso, a base relaciona “Administrador” e os objetos geo-referenciados “Cidade”, “Estado”, “País”, indicando que “Administrador” pode gerenciar estes objetos espaciais nos papéis de “prefeito”, “governador” ou “presidente”.

A base de conhecimentos é utilizada para: (a) verificar se as entidades espaciais e não espaciais citadas na consulta fazem parte do banco de dados; (b) servir de fonte para a construção de relacionamentos que envolvam objetos não espaciais e determinar o contexto de uso de operadores topológicos; e (c) servir de fonte de informação para as meta-consultas, isto é, consultas sobre o esquema do banco de dados.

3.5 O Gerador de Consultas em LEGAL

O gerador é utilizado na parte final da tradução. Ele é responsável pela construção da consulta em LEGAL. Problemas típicos dessa fase são a qualificação de atributos para eliminar ambigüidades e a utilização de *aliases*. Quando um atributo relativo à cláusula “SELECT” ou “WHERE” pertence a mais de uma relação citada em “FROM”, o gerador de consultas em LEGAL qualifica, através da pre-fixação, esse atributo.

Por exemplo, a consulta “Qual é a população média das cidades cujo prefeito é do partido PT?” gera os seguintes termos em LEGAL

```
SELECT populacao
FROM c IN cidade, a IN administrador
WHERE partido = ‘PT’ AND prefeito = a.nome
```

onde o atributo “nome” é prefixado, já que ele pertence a ambas as relações envolvidas na consulta. Por outro lado, a consulta “Quais são as rodovias que cruzam a rodovia Dutra e que cruzam a estrada BR265?” daria origem aos seguintes termos em LEGAL

```
SELECT e1
FROM e1 IN estrada, e2 IN estrada, e3 IN estrada
WHERE cross(e1,e2) AND e2.nome = ‘Dutra’ AND
      cross(e1,e3) AND e3.nome = ‘BR265’
```

onde são utilizados os *aliases* “e1”, “e2” e “e3”. O operador espacial “cross” utilizado na consulta se aplica à condição de cruzamento de rodovias porque, na representação armazenada, trata-se de duas linhas poligonais. Caso as representações fossem outras, o verbo “cruzar” deveria ser traduzido em outro tipo de operação em LEGAL. Este tipo de problema será discutido em mais detalhes na seção 5.

4 Problemas Linguísticos

A maioria dos problemas linguísticos enfrentados pelo tradutor são comuns a qualquer aplicação em LN. Um tradutor de LN não deve processar somente sentenças isoladas; a referência a objetos mencionados em sentenças anteriores é uma construção muito comum na língua natural e permite que as consultas sejam feitas de maneira mais natural e, portanto, mais próxima da comunicação efetuada pelos seres humanos. A elipse e a referência pronominal são exemplos dessas construções.

4.1 Elipse e Referência Pronominal

Elipse é a omissão de um termo da oração facilmente identificável através de elementos presentes na própria sentença, ou através do contexto [Cega91]. Algumas formas de elipse são resolvidas a partir da estrutura gramatical da sentença. Quando o termo omitido é um atributo relativo à cláusula “WHERE”, procura-se na própria sentença por palavras que se refiram a esse termo. Quando o termo omitido é relativo a uma relação ou a atributos relativos à cláusula “SELECT”, recorre-se ao contexto na busca pelo termo elidido.

Assim como a elipse, a referência pronominal também abrevia as sentenças tornando-as mais adequadas a uma interação mais natural. Somente um tipo de referência pronominal é tratada neste trabalho: a referência à consulta anterior. As seguintes consultas processadas pelo protótipo mostram como a combinação de elipse e referência pronominal torna a interação com o usuário mais natural e menos cansativa:

(a) Qual é o nome e a área das cidades com população inferior a 55000 habitantes?

```
SELECT nome, area  
FROM c IN cidade  
WHERE população < 55000
```

(b) E das cidades a menos de 50 km do rio Tietê?

```
SELECT c.nome.area  
FROM c IN cidade, r IN rio  
WHERE distance(c,r) < 50 AND r.nome = “Tietê”
```

(c) E qual a altitude dessas cidades?

```
SELECT altitude  
FROM c IN cidade, r IN rio  
WHERE distance(c,r) < 50 AND r.nome = “Tietê”.
```

Na consulta (c) a referência pronominal “dessas cidades” é resolvida através do contexto, obtido da sentença anterior; a elipse da consulta (b), por sua vez, é resolvida através da sentença (a).

4.2 Ambigüidade

A ambigüidade é um dos problemas mais difíceis de serem resolvidos, podendo ser de diferentes tipos. Por exemplo, na sentença “A manga é amarela”, o substantivo “manga” tanto pode significar uma fruta como parte de uma roupa. Esse tipo de ambigüidade pode ser eliminado em situações de domínio restrito; se o banco de dados considerado for sobre flora, “manga” estará se referindo a fruta. Mesmo considerando-se um domínio restrito, podem ocorrer ambigüidades.

O tradutor tenta resolver automaticamente as ambigüidades encontradas e, caso não consiga, solicita ao usuário que indique a opção desejada. Por exemplo, a sentença “Qual é a altitude da cidade cujo estado tem população maior que 100000 e cuja área é inferior a 100000 m²?” é ambígua, visto que o atributo área tanto pode pertencer à Cidade ou a Estado. Aqui, é preciso recorrer ao usuário para saber a qual dos dois ele está se referindo. Já na sentença “Qual é a cidade cujo estado tem área superior a 500000 m² e altitude inferior a 100 m?” não há necessidade de consultar o usuário. Embora ele não tenha especificado a que relação o atributo altitude pertence (caracterizando, portanto, uma pergunta incompleta e ambígua), somente Cidade possui esse atributo. Portanto, o tradutor faz o relacionamento desse atributo corretamente.

5 Consultas Espaciais e Representações Múltiplas

Os seguintes tipos de consulta são tratadas pelo tradutor: (a) consultas envolvendo relacionamentos espaciais; (b) consultas envolvendo negação de termos que expressam relacionamentos espaciais; e (c) disjunção de expressões que envolvem relacionamentos espaciais. Consultas envolvendo relacionamentos espaciais podem utilizar basicamente tres tipos de operações: métricas, topológicas e de direção. O LEGAL permite tratar dos dois primeiros tipos, sendo que o terceiro ainda não é suportado adequadamente pelos SIGs comerciais.

Do ponto de vista de processamento de consultas espaciais, existem duas considerações a serem feitas:

- A eficiência do processamento destas consultas depende da existência prévia de índices espaciais armazenados no banco de dados. Este tipo de problema foge ao escopo deste trabalho, já que o desempenho da consulta depende da implementação interna do SIG (no caso, de LEGAL) e portanto o tradutor não pode se ocupar disto.
- Os operadores espaciais a serem utilizados dependem da representação armazenada. Este ponto, já mencionado anteriormente neste artigo, será tratado a seguir.

5.1 Consultas envolvendo operações métricas

O processamento de operações métricas (distância, perímetro etc) é, como mencionado anteriormente, dependente da representação. O exemplo da distância entre vários municípios e Belo Horizonte, na seção 2, mostra este tipo de problema. Se todos os municípios estiverem armazenados em imagens, a distância precisa ser calculada de uma determinada forma; se a escala de armazenamento for pequena, os municípios podem ser considerados como pontos; se a escala de armazenamento for maior, a distância exige cálculo do centróide de cada polígono que circunscribe o município, e assim por diante. Havendo múltiplas representações, o LEGAL exige que o usuário indique sobre qual representação as operações métricas devem ser calculadas. No caso do tradutor implementado, foi realizada uma simplificação de forma a associar a cada objeto espacial uma única representação: Estado, País e Lagos foram representados como *polígonos*; Cidade e Município como *pontos*; e Rios e Estradas como *poli-linhas*. Esta mesma simplificação foi usada no processamento de consultas envolvendo operadores topológicos, mas com outras implicações.

5.2 Consultas envolvendo operadores topológicos

O processamento de consultas envolvendo operadores topológicos também depende da representação, mas num grau diferente. Enquanto que uma dada operação (por exemplo, distância) solicitada em LN é traduzida em um único operador de LEGAL (**distance**), cuja implementação depende da representação selecionada, um mesmo vocábulo em LN pode ter *diferentes traduções* em LEGAL quando se trata de operadores topológicos.

O problema, neste caso, é que a escala determina qual o operador a aplicar para um mesmo conceito. Estudos teóricos sobre operadores topológicos [CSE94] permitiram determinar que todos os conceitos de relacionamentos topológicos binários entre dois objetos em um espaço bidimensional podem ser expressos através de 5 operadores (“cross”, “in”, “disjoint”, “touch” e “overlap”, representando, respectivamente, os conceitos de “atravessa”, “inclusão”, “disjunção”, “adjacência” e “sobreposição”).

No entanto, a implementação desses conceitos não é intuitiva. Por exemplo, a consulta “*Quais estradas atravessam a cidade de Belo Horizonte?*” é traduzida em LEGAL, usando o banco de dados exemplo, por:

```
SELECT e
FROM e IN estrada, c IN cidade
WHERE c IN e AND cidade.nome = “Belo Horizonte”.
```

Já a consulta “*Quais estradas atravessam o estado de São Paulo*” é traduzida por:

```
SELECT e1
FROM e1 IN estrada, e2 IN estado
WHERE e1 CROSS e2 AND estado.nome = “São Paulo”.
```

Note que o vocábulo “atravessa” corresponde no primeiro caso a uma tradução de operador topológico “IN” (c IN e), ao invés do operador topológico “CROSS” de LEGAL. A razão é que o operador “IN” foi utilizado porque, na representação do exemplo, Cidades são pontos e Estradas são poli-linhas. Na segunda consulta, praticamente idêntica à primeira, o operador “CROSS” é corretamente utilizado. Este mesmo tipo de problema ocorre em várias outras situações. Por exemplo, a noção de adjacência depende da forma de armazenamento e muitas vezes deve ser traduzida a partir do cálculo de distância, onde duas entidades podem ser consideradas adjacentes se a distância medida em uma representação estiver dentro de um certo limite de tolerância.

O tradutor implementa corretamente estes casos, para as representações especificadas. Há, igualmente, a possibilidade de permitir processamento com representações múltiplas, mas para isso é preciso modificar ligeiramente alguns dos módulos implementados.

5.3 Consultas mistas e com negação

O tradutor permite igualmente resolver o processamento de consultas envolvendo mais de um tipo de operador (espacial e não espacial). Por exemplo, a consulta “*Recupere as cidades cortadas pelo rio Paraopeba e a menos de 100 km da cidade de Belo Horizonte*” envolve operadores métricos e topológicos, sendo transformada em uma consulta aninhada em LEGAL. O processamento interno deste tipo de consulta exige sua divisão em vários blocos de execução, um para cada tipo de operador.

Outro problema é o processamento envolvendo negação ou disjunção, cujo tratamento é muito caro. Embora o tradutor possa processar corretamente tais consultas em LEGAL, a questão da implementação interna não pode ser ignorada. Os detalhes de implementação deste tipo de consulta em LEGAL são discutidos em [CCH+96].

6 Trabalhos correlatos

A pesquisa descrita neste trabalho combina resultados nas áreas de LN, bancos de dados espaciais e SIGs. Esta combinação é, por si só, original.

Existem inúmeros trabalhos desenvolvidos na área de processamento de LN no contexto de bancos de dados. Por exemplo, interfaces em linguagem natural para aplicações convencionais em bancos de dados relacionais são descritas em [And92]; [And96] apresenta o uso de interfaces em linguagem natural para bancos de dados temporais. Os diferentes paradigmas de interfaces para SIG e as várias propostas de linguagens de consulta estão descritas em [OM96] (combinando pesquisas em SIGs e bancos de dados). No entanto, a utilização de LN no contexto de SIG não tem nenhuma implementação discutida na literatura.

Há algum tempo já se sabe que, do ponto de vista do usuário, linguagens relacionais não são adequadas para realizar consultas espaciais (vide por exemplo [Ege92]). A idéia de que a interface deve se aproximar dos padrões mentais do usuário vem motivando todo um conjunto de pesquisas que combinam trabalhos em projeto de interface e ciências cognitivas (por exemplo, [Kuh91, HCS93, Her94]).

Uma primeira tentativa para resolver o problema é a que buscou estender linguagens textuais de consulta de bancos de dados com operadores espaciais e de controle de apresentação. Exemplos nesta direção são a linguagem GEOQL de [Ooi90], que estende o SQL com operadores espaciais e a linguagem SPATIAL-SQL de [Ege94] que, além da parte espacial, provê sintaxe para definição da apresentação do resultado. LEGAL é outro exemplo deste tipo de linguagem. No entanto, estas linguagens sofrem todas do problema de exigirem do usuário conhecimento da estrutura interna do banco de dados espacial e das representações dos dados.

Outra direção de pesquisa visa desenvolvimento de linguagens visuais, como por exemplo [DC91, VMS+93]. O problema, neste caso, é que uma mesma consulta expressa de forma gráfica pode não apenas provocar diferentes traduções para o SIG subjacente, mas também dar margem a diversas interpretações semânticas por parte do usuário.

Finalmente, outro tipo de tentativa de diminuir o gap semântico entre usuário e o SIG envolve o desenvolvimento de interfaces multimodais, que permitem vários modos de interação, inclusive através de LN. Exemplos deste tipo de ambiente são descritos em [EF88, AKK94].

Vale ressaltar que, na literatura correlata, o uso de LN é apenas sugerido (principalmente no caso de interfaces multimodais), mas não há relatos de sua implementação.

7 Conclusões e extensões

Este trabalho apresentou um tradutor de LN para uma linguagem de consulta espacial específica – LEGAL, discutindo as implicações tanto do ponto de vista de LN quanto do ponto de vista de bancos de dados espaciais. Suas principais contribuições são de dois tipos: do ponto de vista de pesquisa, representa combinação de pesquisas em duas áreas da Computação – bancos de dados e LN – para resolver problemas de SIGs; e do ponto de vista prático, valida o estudo através da implementação de um protótipo que é baseado em uma linguagem de consulta espacial utilizando um sistema comercial.

O uso de LN visa aumentar o poder de expressão semântica de uma linguagem de consulta espacial (no caso, LEGAL). Apesar do caráter exploratório do trabalho, percebe-se a grande potencialidade do tradutor, principalmente em se tratando de usuários de SIG. Uma das vantagens observadas é que o usuário, usando este tipo de interação, não mais precisa ter noções relativas ao uso adequado de operadores topológicos em função da representação adotada.

Outra grande facilidade é a construção automática de consultas complexas, incluindo relacionamentos entre entidades espaciais, e não espaciais. Para um usuário leigo em computação seria difícil a criação de tais relacionamentos, já que muitos têm dificuldade em expressar consultas em SQL. O tratamento de sentenças envolvendo elipse e referência pronominal possibilita uma interação menos cansativa, podendo o usuário omitir termos que podem ser obtidos através do contexto ou de informações obtidas na própria sentença.

Apesar das vantagens, a interação em LN proporciona também desvantagens, como a opacidade da cobertura linguística, o desconhecimento da cobertura do domínio, a ambigüidade inerente à LN e a dificuldade de se alcançar a portabilidade em relação a outros domínios. Além disso, a implementação de um tradutor para LN é bastante trabalhosa, especialmente no que diz respeito à definição e implementação das regras gramaticais. Ademais, o tratamento de fenômenos lingüísticos e a conversão de palavras para operadores espaciais não é trivial, dada a dependência do contexto e representação dos dados.

Existem várias extensões ao trabalho apresentado. Uma primeira linha de pesquisa visa o desenvolvimento de uma interface multimodal para SIGs, onde o protótipo desenvolvido seja um dos módulos. Uma outra extensão se refere à solicitação da *apresentação* dos dados. Várias linguagens de consulta espacial, inclusive LEGAL, permitem definir não apenas o resultado desejado, mas também a forma de apresentação (por exemplo, tipo de mapa, cor, preenchimento, legendas). Este tipo de facilidade exigiria estender o vocabulário utilizado e traria maior dificuldade para o interpretador.

Referências

- [And92] I. Androutsopoulos. *Interfacing a Natural Language Front-End to a Relational Database* Msc. Dissertation. Dept. of Artificial Intelligence, University of Edinburgh, 1992.
- [And96] I. Androutsopoulos. *A Principled Framework for Constructing Natural Language Interfaces to Temporal Databases*. Ph.D Thesis. Dept. of Artificial Intelligence, University of Edinburgh, 1996.
- [AKK94] M. Arikawa, H. Kawakita, and Y. Kambayashi. Dynamic Maps as Composite Views of Varied Geographic Database Servers. In *First International Conference, Applications of Databases, ADB94*, 1994.
- [ART93] I. Androutsopoulos, G. D. Ritchie, P. Thanisch. MASQUE/SQL - An Efficient and Portable Natural Language Query for Relational Databases. In *Proc. 6th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*, 1993.
- [Cam95] G. Camara. *Modelos, Linguagens e Arquiteturas para Bancos de Dados Geográficos*. PhD thesis, INPE, dezembro 1995.

- [CCH⁺96] G. Camara, M. Casanova, A. Hemerly, G. Magalhaes, and C. Medeiros. *Anatomia de Sistemas de Informação Geográfica*. 10 Escola de Computação, 1996.
- [Cega91] D. P. Cegalla. *Novíssima Gramática da Língua Portuguesa*. Companhia Editora Nacional, 1991.
- [CM87] W. F. Clocksin, C. S. Mellish. *Programming in PROLOG*. Springer-Verlag, 1987.
- [CSE94] E. Clementini, J. Sharma, and M. Egenhofer. Modelling Topological Spatial Relations: Strategies for Query Processing. *Computer & Graphics*, 18(6):815-822, 1994.
- [DC91] M. Maingenaud D. Calcinelli. The Management of the Ambiguities in a Graphical Query Language for GIS. In *Proc. 2nd Symposium Spatial Database Systems*, pp 99-118, 1991.
- [EF88] M. Egenhofer and A. Frank. Designing Object-oriented Query Languages for GIS: Human Interface Aspects. In *Proc 3rd International Symposium on Spatial Data Handling*, pp 79-98, 1988.
- [Ege92] M. Egenhofer. Why not SQL! *International Journal of Geographical Information Systems*, 6(2):71-86, 1992.
- [Ege94] M. Egenhofer. Spatial SQL: A Query and Presentation Language. *IEEE Transactions on Knowledge and Data Engineering*, 6(1):86-95, 1994.
- [HCS93] S. Hirtle, T. Crippa, and M. Spring. The Cognitive Structure of Space: an Analysis of Temporal Sequences. In *Proc. European Conference Spatial Information Theory - COSIT*, Springer Verlag Lecture Notes in Computer Science 716, pp 177-189, 1993.
- [Her94] D. Hernandez. *Qualitative Representation of Spatial Knowledge*, volume 804 of *Lecture Notes in Artificial Intelligence*. Springer Verlag, 1994.
- [Kuh91] W. Kuhn. Are Displays Maps or Views? In *Auto-Carto*, volume 10, pp 588-598, 1991.
- [OM96] J. Oliveira and C. B. Medeiros. User Interface Architectures, Languages and Models in Geographic Databases. In *Proc XI SBDD*, pp 20-42, 1996.
- [Ooi90] B. C. Ooi. *Efficient Query Processing in Geographic Information Systems*. Springer Verlag Lecture Notes in Computer Science, 1990.
- [PW80] F. C. N. Pereira and D. H. D. Warren. Definite Clause Grammars for Language Analysis - A Survey of the Formalism and a Comparison with Augmented Transition Networks. In *Artificial Intelligence*, vol. 13, pp 231-278, 1980.
- [VMS⁺93] K. Vadaparty, R. Molmen, H. Salem, P. Whiting, and S. Naqvi. Towards a Fully Sheetless GIS with Incremental Querying. In *Proc. ACM/ISCA Workshop on Advances in Geographic Information Systems*, pp 94-99, 1993.
- [Wang94] F. Wang. Towards a Natural Language User Interface: an Approach of Fuzzy Query. In *International Journal of Geographic Information Systems*, 8(2):143-162, 1994.
- [WWW96] Spring online Documentation. In <http://www.inpe.br/spring>.