结合方向关系和拓扑关系的约束满足推理

侯 睿
HOU Rui

湛江师范学院 信息科学与技术学院 计算机系,广东 湛江 524048
Department of Computer Science,College of Information Science and Technology,Zhanjiang Normal University,Zhanjiang,Guangdong 524048,China
E-mail:houcom126@126.com

HOU Rui.Constraint satisfaction reasoning with combining cardinal direction and topology relation.Computer Engineering and Applications,2008,44(16):63-65.

Abstract: Combining Region Connection Calculus (RCC) and cardinal direction based on regions in qualitative spatial reasoning and using the interaction tables for the topology and cardinal direction relation, the algorithm for Constraint Satisfaction Problem (CSP) reasoning combining RCC8 and cardinal direction is proposed, which can be used in spatial reasoning integrating topological and cardinal direction relation.
Key words: topological relation;direction relation;Constraint Satisfaction Problem(CSP);qualitative spatial reasoning

摘 要:结合定性空间推理中的区域连接演算(RCC)和基于区域的主方向关系模型,应用拓扑和方向关系上的复合表,将方向关系和拓扑关系的推理看作约束满足问题(CSP),给出了结合 RCC8 和主方向关系的约束满足问题推理算法,该算法可结合拓扑关系和方向关系进行推理。
关键词:拓扑关系;方向关系;约束满足问题;定性空间推理
DOI:10.3778/j.issn.1002-8331.2008.16.019 **文章编号**:1002-8331(2008)16-0063-03 **文献标识码**:A **中图分类号**:TP18

1 引言

近年来,定性空间推理中空间关系模型的研究取得了很大的进展,这些模型涉及到不同的空间关系:拓扑、方向、形状和距离关系等。在空间关系中拓扑关系和方向关系研究方面,目前已建立了一些形式化的描述模型,然而这些模型都是相对独立的,单纯使用这些独立模型来定性表示空间对象的空间关系、空间分析查询、制图综合等操作的精度还有待提高。现实世界的应用经常需要处理多方面的空间关系,并且由于不同空间关系相互并不独立,单独地处理单一空间关系满足不了实际应用的需要。为了提高模型的描述精度,本文在分析拓扑关系和方向关系理论基础上,提出结合方向关系和拓扑关系的约束满足推理算法。

2 空间拓扑关系表示

目前,拓扑推理的研究主要有两类基本的方法:基于区域连接的 RCC 方法和基于点集的“n-交集”模型。

RCC 方法是基于区域连接的。所谓连接,是指两个区域的闭包共享一点。RCC8 根据区域、区域的闭包和区域的内部,使用 8 个穷举并且不相交的关系来描述两个空间区域之间的拓扑关系。1992 年 Cohn、Cu 和 Randell 基于连通性建立了空间的 RCC 理论[1],得到了 8 种基本的拓扑空间关系,这 8 种拓扑关系分别是不连接(DC)、外部连接(EC)、部分交迭(PO)、相等(EQ)、正切真部分(TPP)、非正切真部分(NTPP)、反正切真部分(TPPI)和反非正切真部分(NTPPI),记 R_8={$DC,EC,PO,TPP,NTPP,TPPI,NTPPI,EQ$}。图 1 描述了区域连接演算 RRC8 的空间对象 x 和 y 的 8 种基本关系。

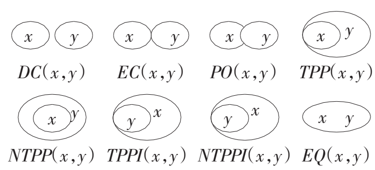

图 1 RCC-8 中的基本拓扑关系

定义 1 设有空间对象 x 和 y,$Top(x,y)$表示目标对象 x 和参考对象 y 之间的拓扑关系是 Top,其中 $Top=\{t|t\in R_8\}$。

描述拓扑关系的代表性关系模型有:基于 RCC 理论的形式化模型和 Egenhofer 等人的 4-交集模型和 9-交集模型等。

3 空间方向关系表示

目前,对于确定性对象间的方向关系模型主要分以点为基元和以区域为基元的两大类模型,另外还有许多学者提出的改进方法。

点对象模型即用抽象点来近似表示空间对象,以点为基元对方向关系建模,在小比例尺空间适用。代表性的点对象模型

作者简介:侯睿(1977-),男,硕士,助教,主要研究领域为 GIS 和 AI。
收稿日期:2008-01-29 **修回日期**:2008-04-21

有“圆锥”模型、投影模型、“双十字”模型,以及对它们的改进及应用研究。

基于区域的方向关系模型考虑到对象的形状及大小对方向关系的影响,代表性模型有 2-D String 方法、最小外接矩形(Minimum Bounding Rectangles,MBR)、主方向关系模型[2]以及 Voronoi 图等。

其中基于区域的主方向关系模型以参考对象为中心,按参考对象的最小边界矩形的四条边将空间划分成 9 个部分(图2),记为 S,SW,W,NW,N,NE,E,SE,O,分别对应地理空间中的南、西南、西、西北、北、东北、东、东南和中心区域,将这 9 个方向关系集合记为 D_9,即 $D_9=\{S,SW,W,NW,N,NE,E,SE,O\}$。通过目标对象与这 9 个部分的交互结果描述目标对象相对于参考对象的方向关系。

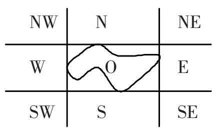

图 2 基于区域的主方位关系模型

定义 2 设有空间对象 x 和 y,$Dir(x,y)$表示目标对象 x 和参考对象 y 的方向关系,x 位于 y 的 Dir 方向,其中 $Dir=\{d|d\in D_9\}$。

4 结合方向关系和拓扑关系的约束满足推理

4.1 空间关系的定性推理

根据空间关系及推理规则的表示,常规推理方法中适合空间关系推理的推理方法[3]有基于谓词逻辑的推理、组合表推理、基于产生式的推理、基于代数的推理和基于语意网络的推理。其中组合表推理和基于语意网络的推理属于基于代数的推理,基于产生式的推理属于基于谓词逻辑的推理。基于组合表的推理是定性空间推理最常用的推理方法。

定义 3 假设 x、y、z 是空间对象,$\Re$ 是定性空间关系的完备集。定性空间推理的基本模式是从已知关系 $R_1(x,y)$与 $R_2(y,z)$推出 $R_3(x,z)$,即

$$R_1(x,y)\circ R_2(y,z)\rightarrow R_3(x,z) \quad (1)$$

其中 $R_1,R_2\in\Re$,$R_3=\{\xi|\xi\in\Re,R_1(x,y)\wedge R_2(y,z)\wedge\xi(x,z)\}$。$R_1(x,y)\circ R_2(y,z)$与 $R_3(x,z)$分别称为推理的前提和结论。

基于 $R_1(x,y)\circ R_2(y,z)$推出 $R_3(x,z)$的过程被称为组合运算,$\circ$ 称为组合运算符。当 $R_1,R_2\in\Re$ 时组合运算是 $\Re\times\Re\rightarrow2^{\Re}$ 的子集称该子集为 $\Re$ 上运算的组合运算表,简称组合表。

例如:对于方向关系,有:

$NE(x,y)\circ NE(y,z)\rightarrow NE(x,z)$

也就是说,如果有空间对象 x、y、z,并且 x 在 y 的 NE 方向,y 在 z 的 NE 方向,可以确定 x 在 z 的 NE 方向。

式(1)给定的前提是 $R_1(x,y)\circ R_2(y,z)$,当修改前提为 $R_1(y,x)\circ R_2(y,z)$时不能从基本推理模式中获得 x 与 z 的关系,为此给出如下关于逆的定义:

定义 4 设 $R\in\Re$,$\Re$ 是定性空间关系的完备集。R 的逆记为 R'指:

$R'=\{\xi|\xi\in\Re$,对任意的空间对象 x 和 y 有 $\xi(y,x)\wedge R(x,y)\}$

根据方向关系的定义,9 种方向关系的逆运算如表 1。

表 1 D_9 的逆关系运算

$Dir(x,y)$	NW	N	NE	W	E	SW	S	SW	O
$Dir'(x,y)$	SE	S	SW	E	W	NE	N	NE	O

根据 R_8 中拓扑关系的定义,8 种拓扑关系的逆运算如表2。

表 2 R_8 的逆关系运算

$Top(x,y)$	DC	EC	PO	TPP	NTPP	TPPI	NTPPI	EQ
$Top'(x,y)$	DC	EC	PO	TPPI	NTPPI	TPP	NTPP	EQ

利用关系的逆可以得到不同推理前提下的推理模式,共有 8 种[4],这 8 种模式穷尽了已知 x 与 y 间的关系和 y 与 z 间的关系经推理获取 x 与 z 间关系的所有情况。

大多数的定性空间推理研究集中在单一的空间方面,而现实世界的应用经常需要处理多空间方面。空间对象的拓扑关系和方向关系并不是相互独立的,只有当区域间的拓扑关系是 DC 和 EC 时,它们之间的方向关系才可能是 8 个(除 O 以外)原子关系中的一种。拓扑关系是结合了方向信息的,例如:区域 x、y 间的拓扑关系若为“包含于”关系,可以推断出 x、y 的方向关系为 O。同样,方向关系有时也提供了拓扑信息,若已知 $Dir(x,y)=W$,则可以知道 $Top(x,y)=DC$ 或者 $Top(x,y)=EC$。

结合多空间方面的研究已得到广泛的认同,但这部分研究工作的进展缓慢:1997 年,Climentini 等人[5]研究了结合方向和距离的定性空间表示与推理;2002 年,Renz 等人[6]提出集成拓扑和距离的空间表示与推理方法,并研究了推理算法的时间代价;2004 年,谢琦等提出了结合时间和位置的时空表示与推理方法[7]。何建华等[8]将拓扑和方向关系推理计算统一到空间目标的 MBR 的比较运算上。许小艳[9]研究了距离和方向关系的组合推理,合理地划分了定性角度值和定性距离值,研究了定性加法,提出了定性减法,并给出了相应的推理模式和规则组合表。

4.2 约束满足问题

约束满足问题作为人工智能领域的研究方向,有大量的计算机领域的问题可以被看作是约束满足问题。例如机器视觉、调度、时态推理、图形处理、建筑的平面图设计、遗传实验设计、满足性问题、电路设计、机械设计与制造以及诊断推理等。一个约束满足问题由以下要素组成:包含 n 个变量 $V_1,\cdots,V_{n-1}$ 的有限集、集合 U(称为问题的全域)和对 U 中可以分配给变量的值的约束的集合,一个约束满足问题是可解的,如果赋给变量 $V_1,\cdots,V_n$ 的值 $a_1,\cdots,a_n\in U$ 满足所有的约束,赋值序列$\{a_1,\cdots,a_n\}$被称为一个解。

约束满足问题作为人工智能领域的研究方向,在经过几十年的研究后,目前已经拥有了数量丰富的各种求解算法,其中搜索算法作为最基本的解决方法,在吸收了一致性算法等新技术后,产生了许多新的搜索算法。而将搜索算法与一致性技术结合起来的算法称为约束传播算法。本文采用约束传播的回跳(Backjumping)搜索算法[10]。

4.3 结合方向关系和拓扑关系的约束满足推理

本文采用的推理方法是形如式(1)的组合表推理。文[11]给出了方向关系和拓扑关系复合运算的 4 个组合表,这 4 个组合表分别是:方向与拓扑进行复合得到拓扑关系的复合表、方向与拓扑进行复合得到方向关系的复合表、拓扑与方向进行复合得到拓扑关系的复合表、拓扑与方向进行复合得到方向关系的复合表。

只要已知空间对象间 x 与 y 以及 y 和 z 的拓扑关系或者是方向关系,利用定性空间推理的模式,根据组合表就能推导出 x 与 z 之间的拓扑关系和方向关系。

结合定性空间推理和基于约束传播算法的回跳搜索,可将

拓扑和方向关系的推理问题描述为如下的 CSP 问题:

设有空间对象集合 $O=\{O_i\}$,$1\leqslant i\leqslant n$,即 O 中共有 n 个空间对象。对象 O_i 和 O_j 的空间关系记为 R_{ij},已知 $O_1,O_2,\cdots,O_m$ 之间的部分拓扑和方向关系,问题是求出其余对象之间的拓扑和方向关系。

定义 5 拓扑和方向关系推理问题定义为约束满足问题 $P(V,D,R)$,其中:

V 是 O 中任意两个对象间的拓扑和方向关系,$V=\{R_{ij}\}$,$1\leqslant i,j\leqslant n(n-1)/2$,$V$ 中共有 $n(n-1)/2$ 个变量;

D 是变量 V 对应的域,分为两部分,对于方向关系是 D_9,而拓扑关系是 R_8;

R 是已知 $O_1,O_2,\cdots,O_m$ 之间的部分拓扑和方向关系的集合,这里用 RT 表示拓扑关系集合,RD 表示对象之间的方向关系集合,即 $R=RT\cup RD$。

问题是求出 V 中未知变量的拓扑和方向关系,也就是求出的解中包含 V 中每个变量的拓扑和方向关系。

这里 V 中共有 $n(n-1)$ 个变量,因为有 n 个空间对象,空间关系不存在对称,每两个空间对象之间存在空间关系,比如对于方向关系有 $Dir(x,y)\neq Dir(y,x)$。但我们只需求出 $Dir(x,y)$,然后根据表 1 就可以知道 $Dir(y,x)$。对于拓扑关系也是如此。因此我们求解的变量个数就是 $n(n-1)/2$。

应用求解约束满足问题的回跳搜索算法求解定义 5 所定义的约束满足问题,由此得到了如下的算法。

算法 求解结合拓扑和方向关系推理算法 TopDirReasoning,简写为 TDR。

输入:n 个空间对象 $O=\{O_i\}$,$1\leqslant i\leqslant n$,一个约束问题 $P(V,D,R)$。

输出:P 的一个解返回或者说明该问题不一致。

步骤:

{(1)//初始化,V_i 为需要赋值的变量,T_i' 和 D_i' 分别为变量 V_i 的拓扑域和方向域

1.　初始化 V_i、T_i'、D_i'、$lastestD_i$、$lastestT_i$

2.　if R 中的变量不一致,return "Inconsistent"

(2)//对 V 中的变量赋值

3.　while $k\leqslant i\leqslant n(n-1)/2$　　//$k-1$ 之前各变量的拓扑和方向关系已知

//给变量赋方位值

4.　{　if　V 中有可赋方向关系值的变量 then　　//寻求能从文[11]方向组合表中查询出方向关系的变量

5.　　{　$D_i'\leftarrow SeekSupport(V_i)$　　//查询复合表获得 V_i 的当前方向域

6.　　$V_i\leftarrow SelectValue(V_i,D_i',lastestD_i)$　　//给当前变量 V_i 赋值

7.　　}

//给变量赋拓扑值

8.　　if V 中有可赋拓扑关系值的变量 then　//寻求能从方向拓扑复合表中查询出拓扑关系的变量

9.　　{　$T_i'\leftarrow SeekSupport(V_i)$ //查询复合表获得当前要赋值变量 V_i 的当前拓扑域

10.　　$V_i\leftarrow SelectValue(V_i,T_i',lastestT_i)$　//给当前变量 V_i 赋值

11.　　}

12.　if　V_i 的赋方向关系值为空或者 V_i 的赋拓扑关系值为空 then

13.　　$i\leftarrow \mathrm{Min}\{lastestT_i,lastestD_i\}$　　//回溯至第 i 个变量

14.　else

15.　　{

16.　　$RD\leftarrow RD\cup\{(V_i,a)\}$　//a 是赋给变量 V_i 的方向关系值

17.　　$RT\leftarrow RT\cup\{(V_i,b)\}$　//b 是赋给变量 V_i 的拓扑关系值

18.　　$i\leftarrow i+1$

19.　　}

20.　}//end while

21.　if $i=k$　//由所给条件无法进行推理

22.　　return "Inconsistent"

23.　else

24.　　return$\{V_1,\cdots,V_n\}$的赋值 R　//返回解

25.　}//end

SelectValue 子过程:

Procedure $SelectValue(V_i,D,lastest_i)$

1.{　while D is not empty

2.　{　取 $a\in D$ 并从 D 中将 a 删除

3.　consistent←true

4.　$k\leftarrow 1$

5.　while $k<i$ and consistent　//将变量 V_i 的赋值 a 与约束集合中的所有元素做一致性检查

6.　{　if $k>lastest_i$ then

7.　　$lastest_i\leftarrow k$　//记录变量 V_i 的死点变量,也即是算法需要回溯的变量

8.　　if $Consistent(\boldsymbol{v}_{i-1},V_i=a)$　//判断 $\boldsymbol{v}_{i-1}$ 与 $V_i=a$ 的一致性,$\boldsymbol{v}_{i-1}$ 表示前 $i-1$ 个变量的赋值

9.　　　$k\leftarrow k+1$　//继续和 $k+1$ 个赋值做一致性检查

10.　　else

11.　　　consistent←false

12.　}//end while

13.　if consistent then //给变量 V_i 赋值成功

14.　　return a

15.　}// end while

16. }//end Procedure

算法 TDR 一开始对变量进行初始化,之后对已知条件进行一致性判定。接下来从第 k 个变量开始赋值,第 4~6 行对 V 中的变量赋方向关系值,$SeekSupport(V_i)$用于方向关系和拓扑关系复合后得到方向关系的复合表[11]中查询出变量 V_i 的方向域,也就是变量 V_i 的方向关系的取值范围。一般情况下,对于方向变量的域是 D_9,对于拓扑关系变量的域是 R_8。从复合表中查询出来的方向域(拓扑域)一般情况会比 $D_9(R_8)$小得多。得到变量的方向域后,就调用过程 $SelectValue(V_i,D_i',lastestD_i)$对变量进行赋值,这里 $lastestD_i$ 表示用于记录对变量进行赋方向关系值不成功时需要回溯的变量位置。对变量赋方向关系值后,第 8~11 行对变量赋拓扑关系值。算法的第 12~19 行对赋值进行判断,如果赋值成功则把成功赋值添加到 RD(方向关系赋值集合)和 RT(拓扑关系赋值集合)中。如果变量 V_i 的方向关系和拓扑赋值有一个失败,则回溯发生,见算法的第 13 行,这里 i 值取 $lastestT_i$ 和 $lastestD_i$ 的最小值,是因为对于赋方向和拓扑关系值时,需要保证赋值的全局一致性,也就是后面变量赋值不成功是由于前面的变量赋值引起的,因此回溯时需要从 $lastestT_i$ 和 $lastestD_i$ 的最小值开始。

$SelectValue(V_i,D,lastest_i)$的第 8 行调用 $Consistent(\boldsymbol{v}_{i-1},V_i=$

(下转 72 页)