Knowledge-Based Systems 22 (2009) 403–409

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Efficient algorithms for spatial configuration information retrieval

Haibin Sun

College of Information Science and Engineering, Shandong University of Science and Technology, Qingdao 266510, China
Hisense state key laboratory of digital multimedia technology, Qingdao 266071, China

ARTICLE INFO

Article history:
Received 26 April 2007
Received in revised form 28 February 2009
Accepted 3 May 2009
Available online 10 May 2009

Keywords:
Spatial reasoning
Constraint satisfaction problem
R-tree
Hash index

ABSTRACT

The problem of spatial configuration information retrieval is a constraint satisfaction problem (CSP), which can be solved using traditional CSP algorithms. But the spatial data can be reorganized using index techniques like R-tree and the spatial data are approximated by their minimum bounding rectangles (MBRs), so the spatial configuration information retrieval is actually based on the MBRs and some special techniques can be studied. This paper studies the mapping relationships among the spatial relations for real spatial objects, the corresponding spatial relations for their MBRs and the corresponding spatial relations between the intermediate nodes and the MBRs in R-tree. Three algorithms are designed and studied, and their performances are compared.

1. Introduction

Spatial configuration retrieval is an important research topic of content-based image retrieval in geographic information system (GIS), computer vision, and VLSI design, etc. A user of a GIS system usually searches for configurations of spatial objects on a map that match some ideal configuration or are bound by a number of constraints. For example, a user may be looking for a place to build a house. He wishes to have a house A north of the town that he works, in a distance no greater than 10 km from his child's school B and next to a park C. Moreover, he would like to have a supermarket D on his way to work. Under some circumstances, the query conditions cannot be fully satisfied at all. The users may need only several optional answers according to the degree of configuration similarity. Of the configuration similarity query problem, the representation strategies and search algorithms have been studied in several papers [1–3,7,8,16,17,21,25,26].

A configuration similarity query can be formally described as a standard binary constraint satisfaction problem which consists of: (1) a set of n variables, $v_0, v_1, \ldots, v_{n-1}$ that appear in the query, (2) for each variable v_i, a finite domain $D_i = \{u_0, \ldots, u_{m-1}\}$ of m values, (3) for each pair of variables (v_i, v_j), a constraint C_{ij} which can be a simple spatial relation, a spatio-temporal relation or a disjunction of relations. In addition, unary constraints such as physical and semantical features can be added to the variables. The goal of query processing is to find instantiations of variables to image objects so that the input constraints are satisfied to a maximum degree. The dissimilarity degree d_{ij} of a binary instantiation $\{v_i \leftarrow u_k, v_j \leftarrow u_l\}$ is defined as the dissimilarity between the relation $R(u_k, u_l)$ (between objects u_k and u_l in the image to be searched) and the constraint C_{ij} (between v_i and v_j in the query). The inconsistency degree can be calculated according to the principles such as conceptual neighborhood [25] or binary string encoding [26]. Given the inconsistency degrees of binary constraints, the inconsistency degree $d(S)$ of a complete solution $S = \{v_0 \leftarrow u_p, \ldots v_{n-1} \leftarrow u_l\}$ can be defined as:

$$d(S) = \sum_{\forall i,j,\, i \neq j,\, 0 \leqslant i,j < n} d_{ij}(C_{ij}, R(u_k, u_l)), \{v_i \leftarrow u_k, v_j \leftarrow u_l\}. \quad (1)$$

Given the defined dissimilarity degree $d(S)$, the similarity degree $sim(S)$, which is not affected by the problem scale and is within the range $[0, 1]$, can be defined as:

$$sim(S) = \frac{n(n-1) \cdot D - d(S)}{n(n-1) \cdot D} \quad (2)$$

where $d(S)$ is the dissimilarity degree of the solution S for a query, n is the number of variables in a query, $n(n-1)$ is the set of constraints between distinct variable pairs (including inverse and unspecified constraints), and D is the maximum dissimilarity degree between two constraint relations. Setting an appropriate minimum value MIN for $sim(S)$ can help to obtain the balance between the approximation degree of the solutions to query conditions and processing cost. The smaller the MIN, the more the solutions obtained, while the processing cost increases too.

In the real world, spatial data often have complex geometry shapes. It will be very costly if we directly to calculate the spatial relationships between them, while much invalid time may be spent. If N is the number of spatial objects, and n the number of

E-mail address: offer_sun@hotmail.com

0950-7051/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.knosys.2009.05.004

query variables, the total number of possible solutions is equal to the number of n-permutations of the N objects: $N!/(N-n)!$. Using minimum bounding rectangles (MBRs) to approximate the geometry shapes of spatial objects and calculating the relations between rectangles will reduce the calculation greatly. So we can divide the spatial configuration retrieval into two steps: firstly the rectangle combinations for which it is impossible to satisfy the query conditions will be eliminated, and then the real spatial objects corresponding to the remaining rectangle combinations will be calculated using computational geometry techniques. To improve the retrieval efficiency, the index data structure which is called R-tree[4]or the variants R+-tree [5] and R*-tree [6] can be adopted.

The next section takes topological and directional relations as examples to study the mapping relationships between the spatial relationships for MBRs and the corresponding relationships for real spatial objects; Section 3 studies three spatial configuration retrieval algorithms; Section 4 presents the experimental system for comparing the three algorithms, designs the experiments, analyzes the experimental results and make a conclusion; the last section concludes this paper.

2. Spatial mapping relationships

This paper mainly concerns the topological and directional relations for MBRs and the corresponding spatial relationships for real spatial objects. The ideas in this paper can be applied to other relationships such as distance and spatio-temporal relations, etc.

2.1. Topological mapping relationships

This paper focuses on RCC8 [9] (see Fig. 1) relations and studies the mapping relationship between the RCC8 relations for real spatial objects and the RCC8 relations for the corresponding MBRs. Let p and q be two real spatial objects, p' and q' be their corresponding MBRs. If the spatial relation between p and q is PO (Partly Overlap), then the possible spatial relation between p' and q' is PO (Partly Overlap) or TPP (Tangential Proper Part) or NTPP (Non-Tangential Proper Part) or EQ (Equal) or TPPi (inverse of Tangential Proper Part) or NTPPi (inverse of Non-Tangential Proper Part) which can be denoted by the disjunction form PO $(p', q') \vee$ TPP $(p', q') \vee$ NTPP $(p', q') \vee$ EQ $(p', q') \vee$ TPPi $(p', q') \vee$ NTPPi (p', q'). To use R-tree to improve the efficiency of the spatial configuration retrieval, the topological relations in the query condition should first be transformed to the corresponding topological relations for the MBRs, which can be used to eliminate the rectangle combinations that cannot fulfill the constraints from the leaf nodes in the R-tree. The intermediate nodes in the R-tree can also be used to fast the retrieval process. Let p' be the rectangle that enclose p', i.e. the parent node of leaf

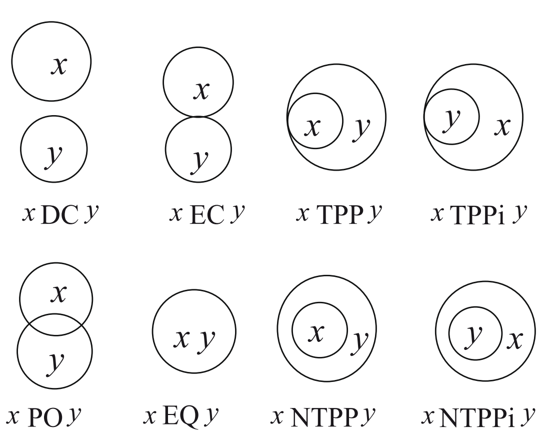

Fig. 1. Two-dimensional examples for the eight basic relations of RCC8.

Table 1
The spatial relations between two real spatial objects, the possible spatial relations that their MBRs satisfy and the possible spatial relations between the corresponding intermediate node and the MBR.

RCC8 relation between p and q	RCC8 relation between MBRs p' and q'	RCC8 relation between p' and q'
DC(p, q)	DC(p', q') ∨ EC(p', q') ∨ PO(p', q') ∨ TPP(p', q') ∨ NTPP(p', q') ∨ EQ(p', q') ∨ TPPi(p', q') ∨ NTPPi(p', q')	PO(p', q') ∨ TPP(p', q') ∨ NTPP(p', q') ∨ EQ(p', q') ∨ TPPi(p', q') ∨ NTPPi(p', q') ∨ EC(p', q') ∨ DC(p', q')
EC(p, q)	EC(p', q') ∨ PO(p', q') ∨ TPP(p', q') ∨ NTPP(p', q') ∨ EQ(p', q') ∨ TPPi(p', q') ∨ NTPPi(p', q')	EC(p', q') ∨ PO(p', q') ∨ TPP (p', q') ∨ NTPP(p', q') ∨ EQ(p', q') ∨ TPPi(p', q') ∨ NTPPi(p', q')
PO(p, q)	PO(p', q') ∨ TPP(p', q') ∨ NTPP (p', q') ∨ EQ(p', q') ∨ TPPi(p', q') ∨ NTPPi(p', q')	PO(p', q') ∨ TPP(p', q') ∨ NTPP(p', q') ∨ EQ(p', q') ∨ TPPi(p', q') ∨ NTPPi(p', q')
TPP(p, q)	TPP(p', q') ∨ NTPP(p', q') ∨ EQ(p', q')	PO(p', q') ∨ TPP(p', q') ∨ NTPP(p', q') ∨ EQ(p', q') ∨ TPPi(p', q') ∨ NTPPi(p', q')
NTPP(p, q)	NTPP(p', q')	PO(p', q') ∨ TPP(p', q') ∨ NTPP(p', q') ∨ EQ(p', q') ∨ TPPi(p', q') ∨ NTPPi(p', q')
TPPi(p, q)	EQ(p', q') ∨ TPPi(p', q') ∨ NTPPi (p', q')	EQ(p', q') ∨ TPPi(p', q') ∨ NTPPi(p', q')
NTPPi(p, q)	NTPPi(p', q')	NTPPi(p', q')
EQ(p, q)	EQ(p', q')	EQ(p', q') ∨ TPPi(p', q') ∨ NTPPi(p', q')

node p' in the R-tree, which is called intermediate node. Given the spatial relation between p' and q', the spatial relation between p' and q' can be derived. For example, from the spatial relation TPP(p', q'), the spatial relation PO (p'', q') ∨ TPP(p'', q') ∨ EQ(p″, q') ∨ TPPi(p″, q') ∨ NTPPi(p'', q') can be obtained. It is very interesting that the parents of the intermediate nodes also have the same property. Table 1 presents the spatial relations between two real spatial objects, the possible spatial relations that their MBRs satisfy and the possible spatial relations between the corresponding intermediate node and the MBR.

Based on the above mapping relationship and the R-tree, the candidate MBR combinations can be retrieved efficiently, and then a refinement step is needed to derive the spatial relations among the real spatial objects that the MBRs enclose, which means that the spatial relation between p and q should be derived from the spatial relation between p' and q'. From the spatial relation between two MBRs, we can derive several possible spatial relations or only one definite spatial relation between two real spatial objects that the MBRs enclose. In the former case, the complex geometry computation will be applied whereas it will be omitted in the latter case. For example, given the spatial relation NTPPi (p', q'), we can derive DC(p, q) ∨ EC(p, q) ∨ PO(p, q) ∨ NTPPi(p, q) ∨ TPPi(p, q), the geometry computation must be adopted to ascertain the spatial relation between p and q. But if we know the spatial relation DC(p', q'), then spatial relation DC(p, q) can be derived directly.

2.2. Direction mapping relationships

According to Goyal and Egenhofer's cardinal direction model [10], there are nine atomic cardinal direction relations (O, S, SW, W, NW, N, NE, E, SE) (see Fig. 2) and totally 218 cardinal direction relations for non-empty connected regions in the Euclidean space $\mathfrak{R}^2$ (illustrated by 3×3 matrix, see Fig. 3) [11].

There are 36 cardinal direction relations for the non-empty and connected regions' MBRs: O, S, SW, W, NW, N, NE, E, SE, S:SW, O:W, NW:N, N:NE, O:E, S:SE, SW:W, O:S, E:SE, W:NW, O:N, NE:E,