Indexing Large Geographic Datasets With Compact Qualitative Representation[*]

Zhiguo Long[†1], Matt Duckham[2], Sanjiang Li[1], and Steven Schockaert[3]

[1]QCIS, FEIT, University of Technology Sydney, Australia
[2]Infrastructure Engineering, University of Melbourne, Australia
[3]School of Computer Science & Informatics, Cardiff University, UK

Abstract

This paper develops a new mechanism to efficiently compute and compactly store qualitative spatial relations between spatial objects, focusing on topological and directional relations for large datasets of region objects. The central idea is to use minimum bounding rectangles (MBRs) to approximately represent region objects with arbitrary shape and complexity and only store spatial relations which cannot be unambiguously inferred from the relations of corresponding MBRs. We demonstrate, both in theory and practice, that our approach requires considerably less construction time and storage space, and can answer queries more efficiently than the state-of-the-art methods.

Keywords: Compact representation; Qualitative spatial reasoning; Region Connection Calculus; Cardinal Direction Calculus; Query answering

1 Introduction

Large volumes of spatial information are continuously being collected from heterogeneous sources. Although most spatial information is stored in a quantitative way, humans often prefer a qualitative approach to describe, interpret, understand and query spatial information. For example, people might ask questions such as ‘Does our street belong to the catchment area of school X?’, ‘Is my neighbourhood within some high-risk area of criminal activities?’, ‘Is the school near a bush fire now in danger as the wind is blowing from north to south?’, and so on. Further, qualitative spatial information can be more flexible—in the sense that it allows answering queries even when precise geometric details are unavailable (e.g., due to privacy reasons or storage limits)—and more timely because precision usually requires more careful and time-consuming measurement.

[*]This is a draft version from the authors. The published version is online at the website of IJGIS. See http://www.tandfonline.com/doi/abs/10.1080/13658816.2015.1104535

[†]Corresponding author: Zhiguo.Long@student.uts.edu.au

The field of qualitative spatial reasoning (QSR) has made significant progress in modeling and reasoning about qualitative spatial relations (Cohn and Renz 2008). However, most current applications of QSR only deal with a relatively small number of variables. QSR does not yet scale to domains with hundreds of thousands of variables because these methods represent spatial information in a qualitative constraint network (Cohn and Renz 2008), the size of which is quadratic (if all relations are binary) or cubic (if some relations are ternary) in the number of variables. An important challenge for the QSR community thus lies in developing methods for representing qualitative spatial information more compactly, such that queries of interest can still be answered efficiently.

Various sources of qualitative information may be considered in applications. Common sources of qualitative spatial relations are descriptions of volunteers and textual descriptions on the web (Goodchild 2007, Hoffart *et al.* 2013). However, we may also derive qualitative representations from geometric information. We will consider this latter possibility, where the main aim of qualitative representations is to improve the efficiency of query answering. Qualitative representations also play an important role in GIS for spatial data adjustment (Wallgrün 2012), human-friendly interaction (Caduff and Egenhofer 2007) and for respecting privacy issues when working with sensitive data.

Given the 2D geometric representation of a large number of regions (e.g. administrative areas), the aim of this paper is to efficiently compute and compactly store the qualitative spatial relations between these regions. Our novel approach is applied to RCC8 relations (Region Connection Calculus) from Randell *et al.* (1992) and CDC relations (Cardinal Direction Calculus) from Goyal and Egenhofer (1997) and Liu *et al.* (2010) for large datasets of region objects. The main idea is to use minimum bounding rectangles (MBRs) to approximately represent spatial region objects with arbitrary shape and complexity, and only store those spatial relations which cannot be unambiguously inferred from the spatial relations between corresponding MBRs. While MBRs have been extensively used in spatial indexing (see Sect. 5), to the best of our knowledge, our approach is the first that considers MBRs for reducing the size of qualitative representations. We prove that our approach can represent qualitative spatial information more efficiently than existing alternatives (in terms of both time and space). Furthermore, we experimentally show that our approach can indeed be used to answer queries more efficiently than state-of-the-art techniques.

The remainder of this paper is organised as follows. After a concise introduction of RCC8, CDC and spatial clustering indices in Section 2, our MBR-based approach and query support are described and theoretically analysed in Section 3 and empirically evaluated in Section 4. Section 5 discusses related work and we conclude and outline future work in Section 6.

2 Background

In this section, we first briefly introduce the qualitative calculi used in this paper, i.e. RCC8 and CDC. The second part of this section describes the spatial clustering index proposed in Fogliaroni (2012), which is perhaps the work most closely related to ours.

In this paper, we represent a spatial object as a bounded *region* in the plane, which is a nonempty regular closed set in the plane. Recall that a regular closed set in the plane is a subset of points which coincides with the closure of its interior, or intuitively, a region that has no one-dimensional parts. The regions we consider could be connected (i.e. one-piece) or disconnected (i.e. consisting of several disjoint pieces).

2.1 Qualitative Spatial Calculi

Suppose U is a domain of spatial or temporal entities. Write $\mathbf{Rel}(U)$ for the Boolean algebra of binary relations on U. A qualitative calculus $\mathfrak{C}$ on U is a finite Boolean subalgebra of $\mathbf{Rel}(U)$. A relation α in a qualitative calculus $\mathfrak{C}$ is *basic* if it is an atom in $\mathfrak{C}$. Well-known qualitative calculi include, among others, RCC8 (Randell *et al.* 1992), and CDC (Goyal and Egenhofer 1997).

The RCC8 algebra is the most influential topological relation model. It has eight basic relations $\mathbf{DC}, \mathbf{EC}, \mathbf{PO}, \mathbf{TPP}, \mathbf{NTPP}, \mathbf{TPP}^{-1}, \mathbf{NTPP}^{-1}$, and $\mathbf{EQ}$, as illustrated in Figure 1 ($\mathbf{TPP}^{-1}$ and $\mathbf{NTPP}^{-1}$ are the inverse of $\mathbf{TPP}$ and $\mathbf{NTPP}$, respectively).

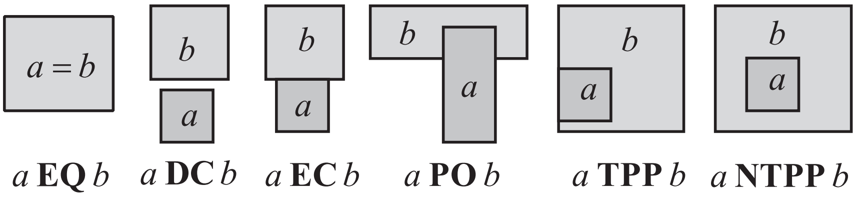


Figure 1: Illustration of RCC8 relations.

CDC is a popular directional relation model. To encode the directional information of a target region a to a reference region b, it will make use of the MBR of b, denoted as $\mathcal{M}(b)$ (see Figure 2 (a) for illustration). By extending the four edges of $\mathcal{M}(b)$, it decomposes the plane into nine tiles, named as NW, N, NE, W, O, E, SW, S, SE (see Figure 2 (b)), and represents the relation $\delta(a, b)$ from a to b as a subset of $\{NW, N, NE, W, O, E, SW, S, SE\}$, where a tile name, say NW, is in $\delta(a, b)$ if and only if an interior point of a is in tile NW (see Figure 2 (b) for illustration). We call $\delta(a, b)$ a single tile relation if it contains only one tile name. For convenience, we also write the tile name in the single tile relation $\delta(a, b)$ for this CDC relation. If only connected regions are considered, then there are 218 basic relations in CDC; if arbitrary bounded regions are considered, then there are 511 basic relations in CDC. See Liu *et al.* (2010) and Goyal and Egenhofer (1997) for more information.

2.2 Spatial Clustering Index

For very large datasets, it is not feasible to represent the RCC8/CDC relations of a spatial geometric dataset with a complete qualitative constraint network, due to the quadratic size of such a network. To reduce the calculation and storage without loss of qualitative information, Fogliaroni *et al.* (2011) and Fogliaroni (2012) have proposed the spatial clustering index, to provide a more compact and efficiently computable qualitative representation. The approach uses so-called *clustering relations* to reduce the calculation and storage of qualitative relations between regions in the dataset. Given a qualitative calculus $\mathfrak{C}$, a relation $\alpha \in \mathfrak{C}$ is a clustering relation if it is downward closed under set inclusion, i.e. from $(a, b) \in \alpha$ and $a' \subseteq a, b' \subseteq b$, we

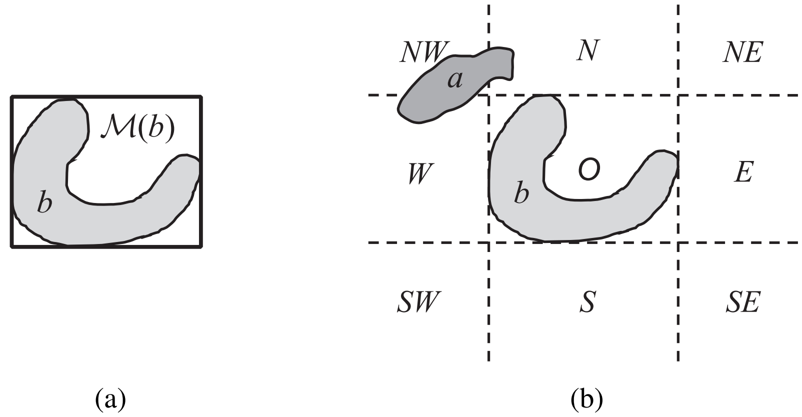


Figure 2: (a) A bounded connected region b and its MBR; (b) The 9 tiles of b and illustration of CDC relation $\delta(a, b) = \{W, NW, N\}$.

infer $(a', b') \in \alpha$. It is straightforward to show that RCC8 only has one clustering relation, viz. **DC**, and CDC has exactly four, viz. the single tile relations NW, NE, SW, and SE.

Given a 2D spatial geometric dataset $\mathcal{D} = \{o_i : i \in I\}$, the approach from Fogliaroni *et al.* (2011) and Fogliaroni (2012) makes use of some auxiliary geometric shapes, called 'index tiles' to help detect clustering relations between regions *associated with* them. Regions are associated with these index tiles according to some predefined strategy for a specific instance of spatial clustering index, such as the intersection between a region and the index tile. In general, it works as follows:

1. Build a spatial clustering structure $\mathcal{I} = \{(t_j, C_j) : j \in J\}$, where each t_j $(j \in J)$ is an index tile and $C_j \subseteq \mathcal{D}$ a cluster of regions associated with t_j by some strategies, and $\{o_i : i \in I\}$ is covered by $\{t_j : j \in J\}$ (i.e. $\bigcup_{i \in I} o_i \subseteq \bigcup_{j \in J} t_j$).

2. For $i \in J$ and $j \in J$, execute the following steps:

   - If $(t_i, C_i) = (t_j, C_j)$, then compute and store the RCC8/CDC relation between every two regions in $C_i = C_j$;
   - If (t_i, C_i) and (t_j, C_j) are different, then compute the relation between t_i and t_j;
     - If the relation between t_i and t_j is a cluster relation, then store this relation for t_i and t_j[1] and continue for the next pair of i, j;
     - If the relation between t_i and t_j is not a cluster relation, then compute and store the relations between every region in C_i and every region in C_j.

Fogliaroni (2012) implements two instances of spatial clustering index, one grid-based and the other R*-tree based. The grid clustering index uses rectangles from a grid that covers the

[1] It is optional whether to store the clustering relation between tiles, depending on whether we need that information or not. Based on different query strategies, we may either not need this piece of information such as for the strategy later used in Experiment 2, or need it for some other strategies like the one in Fogliaroni (2012).

dataset as index tiles, and the R*-tree clustering index uses the MBRs of internal nodes in the R*-tree (Beckmann *et al.* 1990) as index tiles. As R*-trees are only one of many variants of R-trees, the R*-tree clustering index can straightforwardly be extended to methods based on other variants of R-trees (Guttman 1984). In this paper, we will refer to this class of methods as *the R-tree clustering index*.

To build a spatial clustering structure, different strategies can be exploited. For RCC8, the grid clustering index uses the strategy that a region r_i is associated with an index tile t iff $t \cap \mathcal{M}(r_i) \neq \varnothing$; for CDC, it uses the strategy that a region r_i is associated with an index tile t iff $\mathcal{M}(r_i)$ and t have a common interior point. For both RCC8 and CDC, the R-tree clustering index hierarchically builds the clustering index and has two strategies. For leaf level index tiles, in the first strategy, a region r_i is associated with a leaf index tile t iff $\mathcal{M}(r_i)$ and t have a common interior point; in the second strategy, a region r_i is associated with a leaf index tile t iff $\mathcal{M}(r_i)$ is contained in t. For a non-leaf level index tile t, in both strategies of the R-tree clustering index, a deeper level index tile t' is associated with t if t' is contained in t.

As has been observed in Chapters 4 and 6 of Fogliaroni (2012), the spatial clustering index has one important weakness. The performance (the reduction ratio in particular) of the spatial clustering index strongly depends on the quality of the clustering index. A bad clustering index not only results in many repeated considerations of region pairs, but also fails to associate a sufficient number of regions to index tiles that are in clustering relations. Moreover, no general optimal strategy has been found yet to obtain a good cluster index other than repeatedly testing different parameters. This means that the performance of the grid or R-tree clustering indexes cannot be guaranteed. Al-Salman (2014) developed another variant of the spatial clustering index for point objects, which aims to obtain a better cluster by using a more sophisticated clustering strategy (i.e. the density-based approach DBSCAN) and using the concave hull of each cluster as the index tile rather than using the MBR for RCC8 case. This approach stores fewer relations, while it costs more time to construct the representation because of the more complex clustering strategy and the use of the concave hull.

3 The MBR-Based Approach

Given a geographic dataset $\mathcal{D}$, consisting of regions $r_1, \ldots, r_n$, suppose we want to construct a representation from which the RCC8/CDC relation between each pair of regions can be easily obtained. The naive approach, i.e. the complete representation, consists of calculating and storing the RCC8/CDC relation for every pair (r_i, r_j) in $\mathcal{D}$. It leads to a set of $\Theta(n^2)$ relations to compute and store. Furthermore, note that it can be expensive to calculate the relation between two regions of arbitrary shape, as regions in some datasets may have thousands of vertices; e.g., for the dataset Australia-adm2 used in our experiment, a region on average has about 2362 vertices and one region even has 148,488 vertices. In particular, it requires $\mathcal{O}(m \log m)$ time to test intersection for two simple polygons with m vertices (Rigaux *et al.* 2001). Thus the complete representation is extremely time and space consuming. Therefore, our task is to find an efficient way to construct a qualitative representation of $\mathcal{D}$ at a significantly lower cost, while still allowing us to easily obtain the RCC8/CDC relation between any two regions.

The spatial clustering index approach from Fogliaroni *et al.* (2011) and Fogliaroni (2012) has made improvements in terms of both calculation and storage, but, as already mentioned, its performance strongly depends on the quality of the clustering index. The prime network approach (Duckham *et al.* 2014, Li *et al.* 2015) can find and remove all redundant RCC8 relations and hence significantly save storage space, but it presumes that the complete network of spatial relations has been calculated in advance, i.e. it could alleviate the required space for storing the final representation, but not the computation time.

Here we propose another alternative, called *the MBR-based approach*, inspired by the following observations:

1. MBRs can usually be obtained in real-world applications at very low cost, and can be stored linearly with respect to the number of objects involved.

2. While the number of region pairs for which two MBRs have no common (interior) point will depend on the nature of the considered datasets, we found that many real-world datasets contain a large number of such pairs (e.g. most administrative datasets), and hence the MBR of a region will usually only intersect with a small number of the MBRs of other regions.

3. The RCC8 relation between two regions can be unambiguously inferred from the RCC8 relation between their MBRs if the two MBRs have no common point.

4. The CDC relation between two regions can be unambiguously inferred from the CDC relation between their MBRs (and the MBRs of their connected components) if the two MBRs have no common interior point.

5. Calculating the RCC8/CDC relation between two MBRs is much easier and more efficient than calculating the RCC8/CDC relation between two arbitrary regions.

3.1 Algorithm

Algorithm 1 shows the main steps of the MBR-based approach. In Line 2, we first find all pairs (r_i, r_j) such that the corresponding MBRs have a common point, and then calculate the RCC8 relations between such pairs in Lines 3-4. Similarly, we calculate the CDC relations in Lines 5-7. Note that since CDC relations are not closed under inverse (Liu *et al.* 2010), we need to calculate and store both the CDC relation from r_i to r_j and that from r_j to r_i.

The idea of the algorithm is related to the widely used standard MBR pre-processing technique, which is used by GIS systems to filter out candidate answers to queries. However, while the standard pre-processing method only aims to improve the computation time of query answering, we propose to use MBRs to construct a more compact representation of qualitative spatial information. Although this idea is conceptually simple, we show that it outperforms the state-of-the-art methods for constructing qualitative spatial indices, both in theory and in experiments.

3.2 Correctness of the Algorithm

We need to show the correctness of the algorithm. This means, after applying the algorithm to a set of regions $\mathcal{D} = \{r_1, ..., r_n\}$, the RCC8/CDC relation between any two regions r_i and

Algorithm 1: Calculating a compact representation using MBRs.

```
Input: A set of regions $\{r_1, \ldots, r_n\}$.
Output: A compact representation from which RCC8/CDC relations can be derived.
    1 Obtain the MBRs $\{\mathcal{M}(r_1), \ldots, \mathcal{M}(r_n)\}$ of $\{r_1, \ldots, r_n\}$;
    2 For RCC8: find all pairs $(r_i, r_j)$, with $i < j$, such that $\mathcal{M}(r_i)$ and $\mathcal{M}(r_j)$ have a
      common point;
    3 For every such a pair $(r_i, r_j)$:
    4       Calculate and store the RCC8 relation between $r_i$ and $r_j$;
    5 For CDC: find all pairs $(r_i, r_j)$, with $i < j$, such that $\mathcal{M}(r_i)$ and $\mathcal{M}(r_j)$ have a common
      interior point;
    6 For every such a pair $(r_i, r_j)$:
    7       Calculate and store the CDC relation from $r_i$ to $r_j$ and that from $r_j$ to $r_i$;
```

r_j is either stored or can be unambiguously inferred from the RCC8/CDC relation between the corresponding MBRs $\mathcal{M}(r_i)$ and $\mathcal{M}(r_j)$ (and the MBRs of the connected components of the regions). In the following, we show that the relations which are not stored can be inferred from the relations between MBRs.

It is easy to see, as the following proposition shows, that for RCC8, the non-stored topological relations can always be inferred from the topological relations between the corresponding MBRs.

Proposition 1 ((Papadias *et al.* 1995, Li and Cohn 2012)). *Given two (connected or disconnected) regions a and b, if $\mathcal{M}(a)$ $\mathbf{DC}$ $\mathcal{M}(b)$, i.e. $\mathcal{M}(a) \cap \mathcal{M}(b) = \varnothing$, then $a\,\mathbf{DC}\,b$.*

The following proposition shows that for CDC and connected regions, a similar conclusion can be obtained.

Proposition 2. *Given two regions a, b, if a is connected and $\mathcal{M}(a)$ and $\mathcal{M}(b)$ have no common interior point, then $\delta(\mathcal{M}(a), \mathcal{M}(b)) = \delta(a, b)$, i.e. the CDC relation of a to b is the same as that of $\mathcal{M}(a)$ to $\mathcal{M}(b)$.*

This is the case when for example $\delta(\mathcal{M}(a), \mathcal{M}(b))$ is a single tile relation other than O, which includes not only the CDC clustering relations but also some others like N and W, so it is more general than the clustering relation approach of Fogliaroni (2012).

For possibly disconnected regions, however, we cannot always get the correct CDC relation merely from the MBRs of the regions. Figure 3 shows such an example, where we can see that the CDC relation from a to b is $\{NW, NE\}$ but the CDC relation from $\mathcal{M}(a)$ to $\mathcal{M}(b)$ is $\{NW, N, NE\}$. In this case, we need to take the connected components of a into consideration.

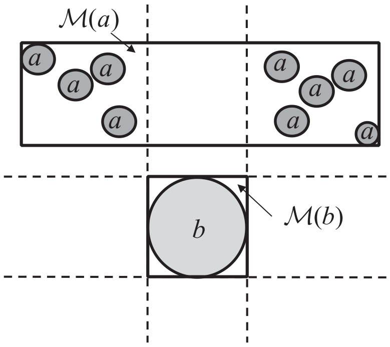


Figure 3: The CDC relation from $\mathcal{M}(a)$ to $\mathcal{M}(b)$ is $\{NW, N, NE\}$, while the CDC relation from a to b is $\{NW, NE\}$.

Lemma 3. *Given two possibly disconnected regions a, b, suppose $a_1, \dots, a_k$ are the connected components of a. Then $\delta(a,b) = \bigcup_{i=1,..,k} \delta(a_i, b)$, where each $\delta(a_i, b)$ is a subset of $\{NW, N, NE, W, O, E, SW, S, SE\}$.*

The above lemma states that the compact representation problem for CDC and possibly disconnected regions can be transformed into the same problem for their connected components. Note that we need to do this only when $\mathcal{M}(a)$ and $\mathcal{M}(b)$ have no common interior point. In this case, we could just compute the CDC relation $\delta(\mathcal{M}(a_i), \mathcal{M}(b))$ for each a_i and take it as the relation $\delta(a_i, b)$, because we know $\mathcal{M}(a_i)$ and $\mathcal{M}(b)$ also have no common interior point. Then we sum them up to obtain the CDC relation between a and b. In summary, for possibly disconnected regions, we have the following conclusion.

Proposition 4. *Given two possibly disconnected regions a, b, suppose $a_1, \dots, a_k$ are the connected components of a, and $\mathcal{M}(a)$ and $\mathcal{M}(b)$ have no common interior point. Then $\delta(a,b) = \bigcup_{i=1,..,k} \delta(\mathcal{M}(a_i), \mathcal{M}(b))$.*

3.3 Efficiency of the Algorithm

The major concern, regarding the efficiency of the algorithm to construct a compact representation, is the number of RCC8 and CDC relations that need to be computed and stored for a given geographic dataset $\mathcal{D}$. For convenience, in this paper we call such number the *qualified size* of $\mathcal{D}$ for the corresponding algorithm (e.g. MBR-based/grid/R-tree clustering indexes), and use it as the measure for the performance of the algorithms.

For grid and R-tree clustering indexes, the qualified size is related not only to the characteristic of the dataset but also to the parameters chosen by the algorithms, such as the index tile size for grid clustering and the maximal number of children allowed in a node of an R-tree. For the MBR-based approach, the qualified size is only related to a particular characteristic of the spatial geometric dataset, i.e. the *average intersection degree* $\bar{d}$ as defined below.

Proposition 5. *Given a spatial dataset $\mathcal{D}$ of n regions. Suppose each MBR m_i (or the interior of* *each MBR for CDC) intersects with d_i other MBRs.*[2] *Let us call $\bar{d} = (\sum d_i)/n$ the* average intersection degree *of $\mathcal{D}$. Then the qualified size of the configuration for the MBR-based approach is $n\bar{d}/2$ for RCC8 relations and $n\bar{d}$ for CDC relations.*

Interestingly, we observe that, for many real-world datasets, most regions only have a relatively small number of 'neighbouring' regions, i.e., the *average intersection degree* tends to be quite small when compared to the number of regions in the configuration. For example, Figure 4 shows the distribution of d_i for the administrative areas of Australia. We have $\bar{d} \approx 7.12$, which is much smaller than the number of regions $n = 1395$. As a result, the qualified size of this dataset for the MBR-based approach will be much smaller than the qualified size for the complete representation.

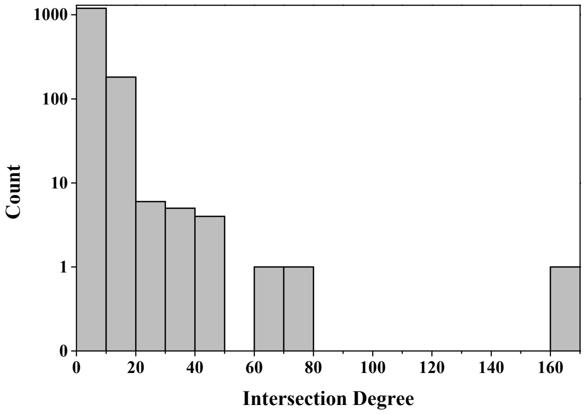


Figure 4: Distribution of intersection degree of the administrative areas of Australia.

Moreover, the qualified size of a given set of regions for the MBR-based approach is never larger than the qualified sizes for the grid or R-tree clustering indexes, for both RCC8 and CDC.

Proposition 6. *Given a set of possibly disconnected regions $\mathcal{D} = \{r_1, ..., r_n\}$, then, for RCC8, the qualified size of $\mathcal{D}$ for the MBR-based approach is at most as large as the qualified sizes of $\mathcal{D}$ for either the grid clustering index or the R-tree clustering index.*

To prove the proposition, we need to show that any relation stored by the MBR-based approach is also stored by both the grid and the R-tree clustering indexes. This is actually true because, for any such relation R_{r_i,r_j}, we can always find two tiles that one contains r_i and the other contains r_j but they are not in a clustering relation. A detailed proof is provided in Appendix A.

A similar result also applies to CDC.

[2] Henceforth we assume the average intersection degree refers to the to slightly different definitions for RCC8 and CDC, respectively, and do not explicitly distinguish them.

Proposition 7. *Given a set of possibly disconnected regions $\mathcal{D} = \{r_1, ..., r_n\}$, for CDC the qualified size of $\mathcal{D}$ for the MBR-based approach is at most as large as the qualified sizes for either the grid or the R-tree clustering indexes.*

The proof of this proposition would be similar to the one for the RCC8 case, with a few differences. We give a detailed proof in Appendix A.

Apart from comparing the qualified size, we also need to compare the preparatory work before the computation of the compact representation of the three algorithms. For the grid clustering index, the preparation is to build the clustering structure. For the R-tree clustering index, the preparation is to build the R-tree. For the MBR-based approach, the preparation is to identify all (interiorly) intersecting MBRs (Lines 1 and 5 of Algorithm 1).

The task of identifying all (interiorly) intersecting MBRs is a classic problem known as the Rectangle Intersection Problem (or Rectangle Spatial Join Problem). There are several well-known efficient algorithms. Using so-called interval trees or segment trees or priority search trees, we can identify all intersecting pairs of MBRs in time $\mathcal{O}(n \log n + k)$, where k is the number of intersecting pairs. For example, see works by Rigaux *et al.* (2001), Bentley and Wood (1980), Güting and Wood (1984) and Güting and Schilling (1987). Since on average there are only a small number of MBRs intersecting a given MBR, the time needed is usually dominated by $\mathcal{O}(n \log n)$. For the datasets used in the experiment, on a computer with Intel$^{®}$ Core$^{\text{TM}}$-i7 3.6 GHz CPU, using brute-force search is already very efficient (less than 50 ms) compared with building the grid clustering structure (about 100 ms and sometimes more than 200 ms) or building an R-tree if we use the efficient bulk-loading STR R-tree by Leutenegger *et al.* (1997) (about 50 ms).

3.4 Query Support

A central function of GIS is to answer spatial queries. Queries about qualitative spatial information include checking the relation between objects and finding instances of regions that satisfy a given spatial constraint (Fogliaroni 2012).

We focus on the former type of queries, which we regard as the most fundamental one. To find all variables that satisfy a relation with a given variable, the essence is to infer or obtain the actual relation between any two variables (i.e. the query type which we focus on), usually after applying some query pre-processing techniques to restrict the search scope (Clementini *et al.* 1994, Papadias *et al.* 1995). In other words, the indexing technique provided by the MBR-based approach can be combined with other query pre-processing techniques to better support queries and real-world applications of qualitative calculi.

To support query answering, our MBR-based approach uses both the stored RCC8/CDC relations and the MBRs of regions (and the MBRs of the connected components of the regions). In general, to identify the relation between two regions, it runs as follows. First, we check if the relation is stored explicitly (e.g., for CDC we check if two MBRs intersect interiorly), if so return it; else use the MBRs to calculate the relation by Proposition 1 and 4.

In practice, we do not need to strictly follow the procedure induced by Proposition 4. Consider the example shown in Figure 3 again. To calculate the CDC relation of a to b, from $\delta(\mathcal{M}(a), \mathcal{M}(b)) = \{NW, N, NE\}$, it is not difficult to see that a must have connected components say a_1, a_2 such that $NW \in \delta(a_1, b)$ and $NE \in \delta(a_2, b)$. Hence we only need to check if there exists a connected component a_i of a such that $N \in \delta(a_i, b)$. Even better, in this case, we only need to check if $I_x(a) = \{x : (\exists y)(x, y) \in a\}$, the x-projection of a, has nonempty intersection with (x_b^-, x_b^+), where

$$x_b^- = \inf\{x : (\exists y)(x, y) \in b\} \quad \text{and} \quad x_b^+ = \sup\{x : (\exists y)(x, y) \in b\} \ . \tag{1}$$

The following lemma provides another observation that can be used to simplify the procedure.

Lemma 8. *Given two possibly disconnected regions a, b such that $\mathcal{M}(a)$ and $\mathcal{M}(b)$ have no common interior point, if $\delta(\mathcal{M}(a), \mathcal{M}(b))$ contains at most two single tile relations, then $\delta(a, b) = \delta(\mathcal{M}(a), \mathcal{M}(b))$.*

Another factor that may affect the efficiency of query answering is the choice of data structures in which the relations (constraints) are stored. One should note that this choice is generally task dependent. For example, we could store the constraints in a relational database, in which case we can use both the variables and relations as identifiers. In the experiments below we will take this approach. It has the advantage of being more flexible in the types of queries that are supported. Note that it is also possible to index the geographic objects by using R-trees and related data structures to improve retrieval efficiency (Guttman 1984, Beckmann *et al.* 1990, Papadias *et al.* 1995). Such techniques mainly focus on answering the type of queries that ask for all instances of regions that satisfy a given qualitative relation. They assume that the qualitative relations are already known and use data structures like R-trees for regions to eliminate instances of regions which could not satisfy the given relation. We should note that the R-tree clustering index used in the experiments is a different technique from the R-tree based techniques discussed here, where the former focuses on building a more compact qualitative representation supporting efficient query answering. Even though these techniques might be useful to further enhance the efficiency of answering the type of queries examined in this paper, we will not consider such optimizations in our experiments, to focus the evaluation on the efficiency of checking spatial relations.

4 Empirical Evaluation

Our approach was evaluated on real-world datasets, in terms of the qualified size and the computation time of query answering. Our implementation makes use of the open source GIS tools GeoTools[3], JTS[4], and the H2 DBMS[5].

4.1 Datasets

Two types of real-world datasets were selected for our evaluation: five datasets about administrative regions from Global Administrative Areas (GADM[6]) based on administrative regions (Real-

[3] http://www.geotools.org/

[4] http://www.vividsolutions.com/jts/JTSHome.htm

[5] http://www.h2database.com/

[6] http://www.gadm.org/

1) and five datasets about environmental habitats from the European Environment Agency[7] (Real-2). These datasets differ considerably in terms of the total numbers of regions and average intersection degree of the MBRs.

Real-1 comprises the following administrative datasets of various sizes: Germany-adm3 (434 regions), Ukraine-adm2 (629 regions), Australia-adm2 (1395 regions), China-adm3 (2411 regions), and USA-adm2 (3145 regions). The five datasets are chosen because of their variation of size. The average intersection degrees of all five datasets are about six to seven. Real-2 contains five datasets of approximately the same number of regions, with different average intersection degrees. In particular, the five datasets of Real-2 were selected to ensure a range of intersection degrees. Each dataset contains around 600 regions, with average intersection degrees of respectively about 45, 106, 122, 180, and 205. The average intersection degree of the full dataset of habitat information is about 426 (about 7.2% of the number of regions), while the five sub-datasets chosen here are 'extreme' cases with average intersection degrees varying from 7.3% to 33.9% of the number of regions in each dataset. We did not use the full dataset because of the following two reasons: (i) that it is hard to collect real-world datasets that have similar size as this one and also have a variation of average intersection degree; (ii) some of the considered baseline methods (e.g. the complete representation) become infeasible when the dataset is this large, though our algorithm can finish in reasonable time.

Both Real-1 and Real-2 contain disconnected regions. The average number of connected components for the regions in the datasets of Real-1 is about 1.2, 1.1, 4.03, 1.88 and 3.50, respectively; that for Real-2 is about 13, 21, 29, 18, and 34, respectively.

4.2 Experiment 1

In Experiment 1, the performance of our algorithm was compared with the grid and R-tree clustering indexes proposed by Fogliaroni (2012), and also with the complete representation.

When applying the grid clustering index, as in Chapter 6 of Fogliaroni (2012) we select a grid $\mathcal{G}$ that covers the dataset such that the size of each index tile in the grid is about the average size of the regions in the dataset $\mathcal{D} = \{r_1, ..., r_n\}$. Following Fogliaroni (2012), we say an index tile t in $\mathcal{G}$ is a 'valid' tile if $t \cap \mathcal{M}(r_i) \neq \varnothing$ for some region r_i; the cluster of regions associated with a valid index tile t is exactly the set of regions r_i such that $t \cap \mathcal{M}(r_i) \neq \varnothing$.

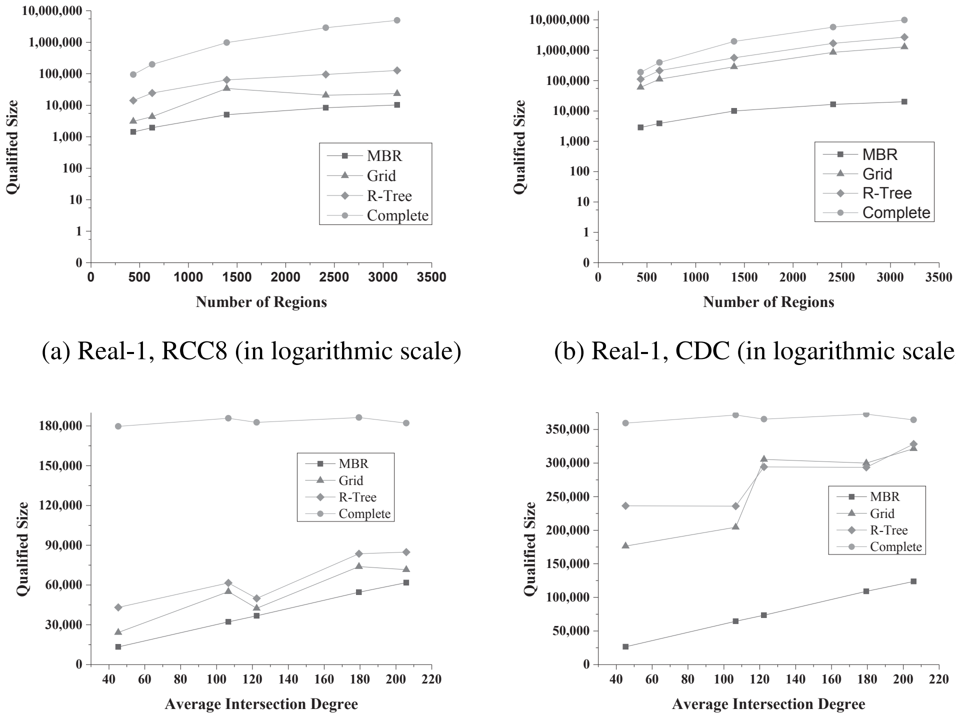
For the R-tree clustering index, since there are no generally accepted optimal parameter settings or tree building strategies, we use the efficient bulk-loading variant of R-tree, the STR R-tree (Leutenegger *et al.* 1997) as an illustration of the performance of the R-tree clustering index. The STR R-tree is designed for static or a priori available objects, which is the case here, and it can efficiently build an R-tree such that only a small number of MBRs overlap. The implementation of STR R-tree used in the experiment is from JTS. We use the default parameter setting of the implementation. Also, we run the R-tree clustering index algorithm from the root level of the tree, because, as suggested in Fogliaroni (2012), the algorithm will reduce more of the qualified size if it starts from a shallower level of the tree.

As noticed in Fogliaroni (2012), the grid clustering index has another weak point. In the grid clustering procedure, some pairs of regions may be simultaneously associated with several

[7] http://www.eea.europa.eu/

index tiles. This will result in these pairs of regions being repeatedly used for computation. If the number of such pairs is large, then the number of qualified relations would be so large that using the grid clustering index would be even more expensive than the complete representation. In our experiments, we check whether a pair of regions has already been considered, before calculating the spatial relation. We have found that the extra work is indeed worth performing, as it can avoid repeated and expensive calculation of qualitative relations between regions.

4.2.1 Results.



(a) Real-1, RCC8 (in logarithmic scale)

(b) Real-1, CDC (in logarithmic scale)

(c) Real-2, RCC8 (in linear scale)

(d) Real-2, CDC (in linear scale)

Figure 5: Qualified size variation with respect to the number of regions and the average intersection degree, on real-world dataset.

Figure 5 shows the results of our experiments for the two real-world datasets. It is immediately clear from Figure 5 that the other algorithms dominate the complete representation ('Complete' in the figures) in all cases. Figures 5(a) and (b) show for Real-1 the qualified size of the MBR-based approach, which is actually $n\bar{d}$ for CDC or $n\bar{d}/2$ for RCC8, and thus grows linearly in the number of regions n for both RCC and CDC relations. The qualified size of the MBR-based approach is consistently smaller than the grid or R-tree clustering indexes. Indeed, the qualified sizes of the grid and the R-tree clustering indexes quickly become prohibitively high as the number of regions increases for CDC (note that the y-axis is in $\log_{10}$ scale), although these two algorithms perform better for RCC8 than for CDC. This is probably because the clustering index mainly helps to distribute disjoint objects into different clusters, which can distinguish the cluster relation $\mathbf{DC}$ clearly from other relations for RCC8 but cannot distinguish well the cluster relations and non-cluster relations for CDC.

The results on Real-2 (Figures 5(c) and (d)) further show that the MBR-based approach is outperforming the other algorithms, and that it linearly depends on the average intersection degree. Taking a closer look, we can see that there are indeed several differences from the results for Real-1. First, the difference between the grid clustering index and the R-tree clustering index becomes much smaller for both RCC8 and CDC (noting that the y-axis uses a linear scale in Figures 5(c) and (d)). Second, the growth rates of the qualified size for the grid and the R-tree clustering indexes are not as high as the growth rates for Real-1. The first difference is due to the large number of regions intersecting with each other in Real-2. This makes it especially hard for both grid and R-tree clustering indexes to find good clustering structures. Consequently there is very little difference between the clustering powers of these two approaches. The second difference is due to poor performance of both algorithms. To be specific, as the average intersection degree grows while the number of regions is fixed, the qualified sizes for both grid and R-tree clustering indexes remain large.

4.2.2 Discussion.

In summary, the MBR-based approach outperforms both grid and R-tree clustering indexes for CDC and RCC8. The advantage of the MBR-based approach is especially noticeable for CDC, because the clustering structures cannot distinguish well the clustering relations for CDC. Another advantage of the MBR-based approach is that it linearly depends on the number of regions and the average intersection degree, while the performance of the other two algorithms is dependent on the tuning of specific parameters. We conclude that the MBR-based approach is a simple yet very efficient method to compute and store CDC/RCC8 information in a compact way.

4.3 Experiment 2

In Experiment 2, the performance of answering a basic spatial query was tested using the different representations: the MBR-based algorithm, the grid clustering index, the R-tree clustering index, as well as the complete representation, and using direct geometric computation of relations between given objects. The specific spatial query used here is to find the CDC/RCC8 relation between two given objects. In the analysis that follows, we focus solely on the more challenging CDC relation. For RCC8, each of the MBR-based algorithm, the grid and R-tree clustering indexes only omits the $\mathbf{DC}$ relation, and so the RCC8 query performance is highly predicable and similar across all these methods.

To answer the query for the representation obtained by the MBR-based approach, we use the 'MBR query method', partially derived from the discussion in Sect. 3.4. The details of the MBR query method are as follows. We write R_{12} for the CDC relation from r_1 to r_2 (i.e. r_2 as the reference object) and S_{12} for the CDC relation from $\mathcal{M}(r_1)$ to $\mathcal{M}(r_2)$, where $\mathcal{M}(r_1)$ and $\mathcal{M}(r_2)$ are the MBRs of r_1 and r_2, respectively. Then:

- If the interiors of the two MBRs $\mathcal{M}(r_1)$ and $\mathcal{M}(r_2)$ do not intersect and r_1 is a connected region, then R_{12} is the same as S_{12}.
- If the interiors of the two MBRs $\mathcal{M}(r_1)$ and $\mathcal{M}(r_2)$ do not intersect and r_1 is a disconnected region, then R_{12} and S_{12} can only be 1-tile, 2-tile, or 3-tile relation. We only need to consider two cases.
  - Case 1: if S_{12} is 1-tile or 2-tile relation, then R_{12} is the same as S_{12}.
  - Case 2: if S_{12} is 3-tile relation, then it can only be one of these four cases: $\{NW, N, NE\}$, $\{NW, W, SW\}$, $\{NE, E, SE\}$, and $\{SW, S, SE\}$. The difference between R_{12} and S_{12} lies only in the presence of the 'middle' tile name (e.g. N is the middle tile name of $\{NW, N, NE\}$, and E for $\{NE, E, SE\}$) in them. Therefore, we only need to check if there is a connected component r_1^i of r_1 which intersects with the middle tile interiorly, and if so then $R_{12} = S_{12}$, otherwise R_{12} is S_{12} excluding the middle tile.

Case 1 is from Lemma 8. We have already seen an efficient technique to deal with Case 2, as mentioned before Lemma 8. We compare the endpoints of the projections of the rectangle $\mathcal{M}(r_1^i)$ and the middle tile rectangle on the x/y-axis. For example, for $S_{12} = \{NW, N, NE\}$, we compare the endpoints of the projections of these two rectangles on the x-axis. If the projections have an interior intersection, then $\mathcal{M}(r_1^i)$ must have an interior intersection with the middle tile N (because $\mathcal{M}(r_1^i)$ can only lie in the area covered by tiles NW, N, and NE). If the projections have no interior intersection, then we know $\mathcal{M}(r_1^i)$ does not have an interior intersection with the middle tile N.

Based on the observation above, direct computation can be optimized in the case where the MBRs do not intersect interiorly, by exploiting exactly the same steps for the MBR query method. In the following experiments, we will always apply this optimization to direct computation.

Turning to the grid and R-tree clustering indexes, it should be noted that Fogliaroni *et al.* (2011) and Fogliaroni (2012) did not consider the mechanism to answer the query discussed here (to find the CDC relation between two given objects). There are many possible ways to answer this query. Here, we use the strategy with the assumption that the MBRs are available:

1. First compute the relation between the MBR of the primary object and the MBR of the reference object;
2. If the relation is one of the clustering CDC relations (NW, NE, SW, SE), then the actual relation is also the same relation;
3. If not, then the relation between the objects must have been previously computed and stored in the representation.

The difference between this strategy and MBR query method mainly lies in the case where the MBR relation is not a clustering relation and at the same time it does not contain tile name O (that is, the two MBRs do not intersect interiorly). In such case, this strategy will query the representation for the relation while MBR query method will just use MBRs to compute the relation.

In the experiment, to make the MBR-based and the grid clustering based query methods comparable, we let the MBR query method first check if two MBRs are in a clustering relation. This will only slightly increase the query time of the MBR query method. For all the methods, when it comes to check if two geometries intersect, we will first use the MBRs of the two geometries to pre-test the possibility of intersection.

4.3.1 Results.

We will assume that the calculated relations, MBRs and geometric information are all available in memory. For each method and dataset, the calculated relations are stored in a database that is hash indexed using the identifiers of geometries as keys, e.g. the key for relation R_{ij} is $i \times N + j$, where N is a sufficiently large integer. The experiments have been done on a computer running Windows® 10, with an Intel® Core$^{\text{TM}}$-i7 3.6 GHz CPU and 16 GB memory.

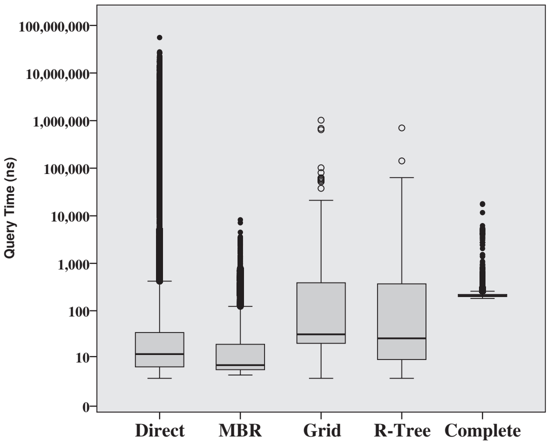


Figure 6: Query answering time of Direct Computation (Direct), MBR query method (MBR), Grid clustering index based query method (Grid), R-tree clustering index based query method (R-Tree), and retrieval from the Complete Representation (Complete). Note that '$\circ$' represents values that lie more than 1.5 box lengths from the hinge of the box.

As we can see from Figure 6, answering queries based on the MBR approach exhibits promising performance. Compared with the two extremes, it is at least as efficient as direct computation from the geometry and is substantially faster than retrieval from the complete constraint network. Like the grid and R-tree clustering indexes, the MBR-based approach represents a compromise that can support more efficient queries than either of these two extremes. In particular, the median query time of the MBR-based approach is respectively about $45\%, 80\%, 76\%$, and 97% lower than the direct computation, the grid clustering index, the R-tree clustering index, and the complete representation. Hypothesis tests also confirm the visual impression from Figure 6, that the average MBR-base approach is more efficient than all the other approaches (t-test, significant at the 1% level). However, such statistical significance is in this case only a guide, as the very large number of samples (200,000 data points) is a known cause of bias towards significance in hypothesis testing. As for the number of I/O operations to the database that stores relations, the MBR-based approach requires about 68% fewer requests than the grid and R-tree clustering indexes, and about 89% fewer than the complete representation. This, to some extent, explains why the MBR-based approach has better performance than these three methods. All of these indicate that the MBR-based approach is a useful alternative in practice to support efficient answering of queries on the qualitative spatial representation, in addition to reducing construction time and size of the representation.

Even though the direct computation shows a performance that is good to some extent, it is worth noting that this method is not always efficient for answering queries. The reason is as follows: There are many 'degenerate' cases for the direct computation. In fact, in the experiment, for the direct computation, 10,645 instances (10% of all the tested ones) have query answering time of more than 10,000 ns, compared with the case of the MBR method which has only 61 instances beyond time 10,000 ns.

Sometimes the calculated relations would be stored in a database on hard disk rather than in memory. In this case, all the methods except for the direct computation will take more time to answer the query, as reading data from hard disk is slower than from memory. The MBR query method will require fewer reading operations than the others because it stores fewer relations. Another possibility is that the MBRs and geographic information are stored on hard disk. In this case, all the methods except retrieving from the complete representation will require more reading operations, as they would probably use the MBRs and geometric information to calculate the relations. The exact impact of storing relations on disk is difficult to measure, however, as performance will crucially depend on optimizations by the database and operating system, such as bundling several queries to the database or by caching some frequent queried relations.

We should also note that although query answering by the MBR query method on average equals or outperforms all the other approaches, it is possible to find or construct degenerate cases, where the performance of this method could be worse than these alternatives. In particular, in the case where the relation of MBRs contain three tile names and these MBRs do not intersect in their interiors, the MBR query method might lead to inefficient queries. In such cases, answering queries by using the MBR query method needs to check if the 'middle' tile name is valid in the real CDC relation. This in turn involves checking intersection of the tile and the MBRs of the connected components in the primary object. For example, in Figure 3, the MBRs of region a and region b are in CDC relation with three tile names $\{NW, N, NE\}$. Note that a contains many connected components and none of them intersect with the middle tile N. In this case, to compute the actual CDC relation between a and b using the MBR approach, the query will probably need to check the MBRs of *all* the connected components of a.

However, such degenerate cases seem to be rare in practice. As we have seen, on average answering queries based on the MBR approach is the most efficient one of all the alternatives tested, because the validity of the 'middle' tile name can usually be confirmed after checking only a few MBRs of connected components. Moreover, the performance on the degenerate cases can be optimized, such as by spatially indexing the MBRs of connected components to reduce the number of tests for intersection.

Finally, we note that all the methods might be further optimized in real-world applications.

For example, the MBRs of connected components of the regions can be indexed using advanced spatial indexing techniques such as R-Tree (Papadias *et al.* 1995) to reduce computation when checking CDC relations. However, we are more interested in the intrinsic performance of the MBR query method itself. Thus, to be clearer about how the MBR query method performs, we were not applying other optimizations, and the experiments here are just simple illustrations of the performance of the MBR query method, to show it is feasible in practice.

5 Alternatives to the MBR-Based Approach

The qualified size of the MBR-based approach can be further reduced. Here, we propose two simple but useful techniques to this end. They are denoted as MBR-C ('C' stands for 'Comparison' as we will see later) and MBR-$\mathbf{DC}$ ('$\mathbf{DC}$' here indicates this technique specifically deals with the RCC8 relation $\mathbf{DC}$). MBR-C works for both CDC and RCC8 representations while MBR-$\mathbf{DC}$ is aimed only at RCC8 representations.

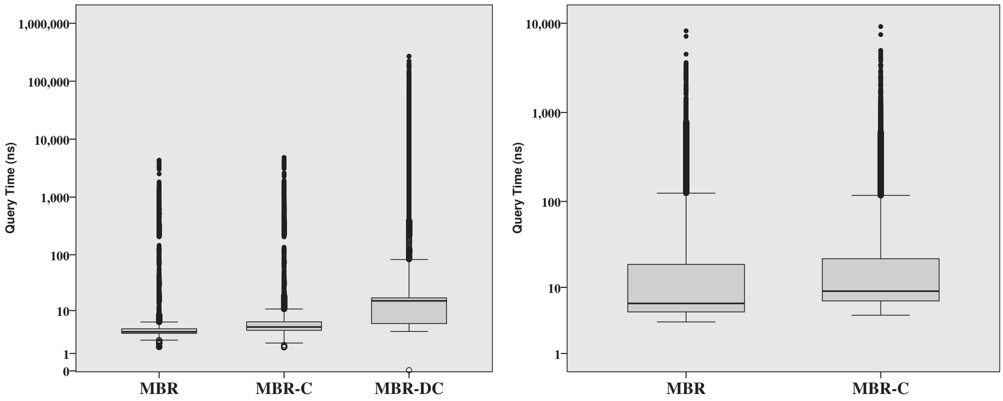
For MBR-C, to decide which relations need to be explicitly stored, we first apply the MBR-based approach, and then compare the CDC/RCC8 relation $R_{a,b}$ between two regions a, b with the CDC/RCC8 relation $R_{\mathcal{M}(a),\mathcal{M}(b)}$ between the MBRs of the two regions $\mathcal{M}(a), \mathcal{M}(b)$. We remove any relation $R_{a,b}$ that is the same as $R_{\mathcal{M}(a),\mathcal{M}(b)}$. This method is based on the assumption that in practice the condition $R_{a,b} = R_{\mathcal{M}(a),\mathcal{M}(b)}$ will usually be satisfied by many pairs of regions. In the query stage, to find the correct relation between a and b, it then suffices to return the relation which is stored, if it is available, and to calculate $R_{\mathcal{M}(a),\mathcal{M}(b)}$ otherwise.

For MBR-$\mathbf{DC}$, we further reduce the qualified size of the RCC8 representation obtained by MBR-C, by removing any $\mathbf{DC}$ relation $R(a, b)$ in the representation that satisfies the following condition: for every connected component a_i of a and for every connected component b_j of b, we have $\mathcal{M}(a_i)\mathbf{DC}\mathcal{M}(b_j)$. To efficiently check this condition, we subsequently check the following conditions:

1. for every connected component a_i of a, we have $\mathcal{M}(a_i)\mathbf{DC}\mathcal{M}(b)$;
2. for every connected component b_j of b, we have $\mathcal{M}(a)\mathbf{DC}\mathcal{M}(b_j)$.
3. for every connected component a_i of a that does not satisfy (1) and for every connected component b_j of b that does not satisfy (2), we have $\mathcal{M}(a_i)\mathbf{DC}\mathcal{M}(b_j)$.

The removed $\mathbf{DC}$ relation $R_{a,b}$ can be retrieved later by using the MBRs of the connected components of the regions: we first check if $R_{a,b}$ is stored, and if not, we check if $\mathcal{M}(a)\mathbf{DC}\mathcal{M}(b)$; if not, we check if one of the above three conditions are satisfied;if not, then we know $R_{a,b} = R_{\mathcal{M}(a),\mathcal{M}(b)}$ by the specification of MBR-C. This method will lead to a smaller representation, based on the assumption that the MBRs of the connected components can better approximate the region, although this will come at the cost of slightly less efficient query answering since a larger number of relations between MBRs may need to be checked. Here, we specifically consider $\mathbf{DC}$ relations because in practice many pairs of regions will be in $\mathbf{DC}$ relation, and in many of these cases, the $\mathbf{DC}$ relation will also hold between the MBRs of the connected components.

Experiments on these approaches show that there is only a slight increase in the time taken to construct the representation, while the qualified size is reduced by a large amount, especially for the CDC case. Figure 7 compares the time of answering queries by these two approaches with that by the original MBR-based approach. On average the original MBR-based approach significantly outperforms both of them (t-test significant at the 1% level). However, in the context of the very large sample sizes, the small effect size is in this case more salient than the statistical significance and to an extent both alternatives are still relatively efficient for answering queries.



(a) RCC8 (b) CDC

Figure 7: Query answering time of MBR-C, MBR-$\mathbf{DC}$ and the original MBR-based approach (MBR).

6 Related Work

MBRs have been used extensively in spatial data structures such as R-trees (Guttman 1984) and their variations such as R*-trees (Beckmann *et al.* 1990) and STR R-tree (Leutenegger *et al.* 1997). In these instances, MBRs are used to index the dataset in order to efficiently answer spatial queries. Guesgen (1989) uses MBRs to represent the Rectangle Algebra (RA) relations between spatial objects. Thanks to their simple representations, MBRs are also used in spatial reasoning to efficiently filter out object pairs that cannot satisfy a particular constraint in a qualitative spatial query. To this end, Clementini *et al.* (1994) and Papadias *et al.* (1995) establish consistent mappings between RA relations and the RCC8 relations for connected regions. For example, it is identified in Papadias *et al.* (1995) that $a\,\mathbf{DC}\,b$ if $\mathcal{M}(a)\,\mathbf{DC}\,\mathcal{M}(b)$ for any connected regions a, b. For possibly disconnected regions, the interaction between RA and RCC8 relations is discussed in Li and Cohn (2012). The interaction between RA and CDC relations is discussed in Liu *et al.* (2010). For a comprehensive discussion of the interaction among RA, CDC, and RCC8 relations, we refer to the recent work in Cohn *et al.* (2014).

Efficient retrieval of qualitative spatial relations has been a hot topic in QSR since the 1990s. Clementini *et al.* (1994) study the use of MBR approximations in query processing involving topological relations. Papadias *et al.* (1995) further consider retrieval of topological relations using spatial data structures like R-trees. Goyal and Egenhofer (2000) also investigate the use of extended cardinal directions in spatial query languages.

In addition to the two variants of the spatial clustering index proposed in Fogliaroni (2012), Al-Salman (2014) studied a variant that clusters the point objects better so that fewer relations are stored, by using the density-based clustering strategy (i.e. DBSCAN) and using the concave hull of cluster as the index tile. The concave hull better approximates the shape of the cluster, but at the cost of increased computation time for constructing the representation.

Finally, some works consider how to remove redundant constraints from a constraint network (Fogliaroni 2012, Wallgrün 2012, Duckham *et al.* 2014, Al-Salman 2014). In particular, the prime network approach (Duckham *et al.* 2014, Li *et al.* 2015) provides an efficient method for removing all redundant qualitative constraints from a complete network. However, such approaches presume that the complete network has somehow been given. This is not the case for our task in this paper, where we are asked to compute and store a compact qualitative representation directly from a set of regions.

7 Conclusion and Future Work

In this paper, we discussed the problem of representing the qualitative spatial relations between a set of regions in more compact ways without loss of information. We proposed a novel and efficient approach for representing the topological RCC8 relations and the directional CDC relations between the regions in a large spatial dataset. This approach has been found to perform well in theory and practice. In particular, in terms of qualified size, our MBR approach is at least as good as the existing approaches (i.e. the complete representation, the grid clustering index, and the R-tree clustering index) in theory and is usually much better in practice. Moreover, our approach is parameter-free, while comparable alternative approaches depend strongly on the chosen parameter values. Future research will consider combining the MBR-based approach with other techniques to improve performance even further. For example, the prime network approach (Duckham *et al.* 2014, Li *et al.* 2015) can be applied to the compact representation to remove redundant constraints; and the relations between MBRs and between regions can be compared to see if they are the same and hence we can safely remove the relations between regions.

Another important problem is to support efficient queries on the qualitative spatial representation. In this paper, for one important type of query (i.e. deriving the RCC8/CDC relation between two given objects), we have provided experimental evidence showing that MBR-based representations lead to queries that are as efficient as any of the alternatives. In fact, the MBR-approach outperforms other approaches in most cases. Our future work on query answering would be to combine the MBR-based approach with other efficient query answering techniques, and examine the support of other types of queries using the representation obtained by the MBR-based approach.

In the current approach, we have not yet considered the possibility of vague or indefinite information. In real-world applications, however, situations in which available information is imperfect abound. Examples include regions with vague boundaries (Papadias *et al.* 1995, Cohn and Gotts 1996) and incomplete information from volunteers or web sources (Goodchild 2007, Winter *et al.* 2011, Hoffart *et al.* 2013). By replacing crisp MBRs with MBRs that have *buffered* boundaries, the MBR-based approach might be able to deal with indefinite information in the dataset. In our subsequent work, we would like to explore this in more detail.

Finally, we will consider further reduction of the qualified size by the MBR-based approach and corresponding efficient support of query answering. The three techniques we briefly discussed in the former section have shown some potential for achieving this.

References

Al-Salman, R., 2014. Qualitative Spatial Query Processing : Towards Cognitive Geographic Information Systems. Thesis (PhD). Universität Bremen.

Beckmann, N., *et al.*, 1990. The R*-tree: An efficient and robust access method for points and rectangles. *In*: H. Garcia-Molina, ed. *Proceedings of the 1990 ACM SIGMOD International Conference on Management of Data*, Atlantic City, New Jersey, USA ACM, NY, USA, 322–331.

Bentley, J.L. and Wood, D., 1980. An optimal worst case algorithm for reporting intersections of rectangles. *IEEE Transactions on Computers*, C-29 (7), 571–577.

Caduff, D. and Egenhofer, M.J., 2007. Geo-Mobile query-by-sketch. *International Journal of Web Engineering and Technology*, 3 (2), 157–175.

Clementini, E., Sharma, J., and Egenhofer, M.J., 1994. Modelling topological spatial relations: Strategies for query processing. *Computers & Graphics*, 18 (6), 815 – 822.

Cohn, A.G. and Gotts, N.M., 1996. The egg-yolk representation of regions with indeterminate boundaries. *In*: P. Burrough and A. Frank, eds. *Proceedings of GISDATA Specialist Meeting on Geographical Objects with Undetermined Boundaries* Taylor & Francis, UK, 171–187.

Cohn, A.G., *et al.*, 2014. Reasoning about topological and cardinal direction relations between 2-dimensional spatial objects. *Journal of Artificial Intelligence Research*, 51, 493–532.

Cohn, A.G. and Renz, J., 2008. Qualitative spatial representation and reasoning. *In*: F. van Harmelen, V. Lifschitz and B. Porter, eds. *Handbook of Knowledge Representation*. Elsevier, Amsterdam, Netherlands.

Duckham, M., *et al.*, 2014. On redundant topological constraints. *In*: C. Baral, G.D. Giacomo and T. Eiter, eds. *Proceedings of 14th International Conference on Principles of Knowledge Representation and Reasoning (KR-2014)* AAAI, California, USA.

Fogliaroni, P., 2012. Qualitative spatial configuration queries – Towards next generation access methods for GIS. Thesis (PhD). University of Bremen.

Fogliaroni, P., *et al.*, 2011. Managing qualitative spatial information to support query-by-sketch. *In*: J. Wang, K. Broelemann, M. Chipofya, A. Schwering and J. Wallgrn, eds. *Proceedings of the COSIT 2011 Workshop Understanding and Processing Sketch Maps* IOS Press, Amsterdam, Netherlands, 21–32.

Goodchild, M.F., 2007. Citizens as sensors: the world of volunteered geography. *GeoJournal*, 69 (4), 211–221.

Goyal, R.K. and Egenhofer, M.J., 1997. The direction-relation matrix: A representation for directions relations between extended spatial objects. *In*: J. Dobson, ed. *The Annual Assembly and the Summer Retreat of University Consortium for Geographic Information Systems Science* UCGIS, NY, USA, 22–81.

Goyal, R.K. and Egenhofer, M.J., 2000. Consistent queries over cardinal directions across different levels of detail. *In*: A. Tjoa, R. Wagner and A. Al-Zobaidie, eds. *11th International Workshop on Database and Expert Systems Applications (DEXA'00)* IEEE Computer Society, USA, 876–880.

Guesgen, H.W., 1989. *Spatial Reasoning Based on Allen's Temporal Logic*. International Computer Science Institute Berkeley.

Güting, R.H. and Schilling, W., 1987. A practical divide-and-conquer algorithm for the rectangle intersection problem. *Information Sciences*, 42 (2), 95–112.

Güting, R.H. and Wood, D., 1984. Finding rectangle intersections by divide-and-conquer. *IEEE Transactions on Computers*, 100 (7), 671–675.

Guttman, A., 1984. R-trees: a dynamic index structure for spatial searching. *In*: B. Yormark, ed. *Proceedings of the 1984 ACM SIGMOD International Conference on Management of Data (SIGMOD '84)* ACM, NY, USA, 47–57.

Hoffart, J., *et al.*, 2013. YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia. *Artificial Intelligence*, 194, 28–61.

Leutenegger, S.T., Lopez, M.A., and Edgington, J., 1997. STR: A simple and efficient algorithm for R-tree packing. *In*: B. Werner, ed. *13th International Conference on Data Engineering* IEEE Computer Society, USA, 497–506.

Li, S. and Cohn, A.G., 2012. Reasoning with topological and directional spatial information. *Computational Intelligence*, 28 (4), 579–616.

Li, S., *et al.*, 2015. On Redundant Topological Constraints. *Artificial Intelligence*, 225, 51–78.

Liu, W., *et al.*, 2010. Reasoning about cardinal directions between extended objects. *Artificial Intelligence*, 174 (12-13), 951–983.

Papadias, D., *et al.*, 1995. Topological relations in the world of minimum bounding rectangles: A study with R-trees. *In*: M. Carey and D. Schneider, eds. *Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data (SIGMOD'95)* ACM, NY, USA, 92–103.

Randell, D.A., Cui, Z., and Cohn, A.G., 1992. A spatial logic based on regions and connection. *In*: B. Nebel and C. Rich, eds. *Proceedings of the 3rd International Conference on Principles of Knowledge Representation and Reasoning (KR-92)* Morgan Kaufmann, Massachusetts, USA, 165–176.

Rigaux, P., Scholl, M., and Voisard, A., 2001. *Spatial Databases: With Application to GIS*. Morgan Kaufmann, Massachusetts, USA.

Wallgrün, J.O., 2012. Exploiting qualitative spatial reasoning for topological adjustment of spatial data. *In*: I.F. Cruz, C.A. Knoblock, P. Krger, E. Tanin and P. Widmayer, eds. *Proceedings of the 20th International Conference on Advances in Geographic Information Systems (SIGSPATIAL'12)* ACM, NY, USA, 229–238.

Winter, S., *et al.*, 2011. Location-based mobile games for spatial knowledge acquisition. *Cognitive Engineering for Mobile GIS*, 780, 1–8.

A Proofs of Proposition 6 and Proposition 7

Proposition 9 ((**Proposition 6**)). *Given a set of possibly disconnected regions $\mathcal{D} = \{r_1, ..., r_n\}$, then, for RCC8, the qualified size of $\mathcal{D}$ for the MBR-based approach is at most as large as the qualified sizes of $\mathcal{D}$ for either the grid clustering index or the R-tree clustering index.*

Proof. Suppose the RCC8 relation for two regions $r_i, r_j \in \mathcal{D}$ is calculated and stored by the MBR-based approach. Then their MBRs must intersect, i.e. $\mathcal{M}(r_i) \cap \mathcal{M}(r_j) \neq \varnothing$. Note that for the grid and the R-tree clustering indexes, the MBRs of objects are used to build the spatial clustering structure, and by the definition of the spatial clustering structure, the tiles in the index completely cover the objects (and, hence, their MBRs), i.e. $\bigcup_{k=1}^{n} \mathcal{M}(r_k) \subseteq \bigcup_{l \in J} t_l$.

For the grid clustering index, by the above assumption, we know there must be one index tile t_0 s.t. $t_0 \cap \mathcal{M}(r_i) \cap \mathcal{M}(r_j) \neq \varnothing$. By the grid clustering indexing strategy of building a clustering structure for RCC8, we know that r_i and r_j are associated with the same clustering index entry (t_0, C_0). Therefore, the RCC8 relation between r_i and r_j will be computed and stored by the grid clustering index.

For the R-tree clustering index, we only need to prove that there exist two leaf level index tiles t_1, t_2 such that $t_1 \cap t_2 \neq \varnothing$ and r_i and r_j are associated with t_1 and t_2 respectively.

As mentioned in the previous section, there are two strategies for the R-tree clustering index to associate a region to a leaf index tile. In the first strategy, by the above assumption, we know that $\mathcal{M}(r_i)$ and $\mathcal{M}(r_j)$, as well as $\mathcal{M}(r_i) \cap \mathcal{M}(r_j)$, are covered by several leaf index tiles. Then among the index tiles that cover $\mathcal{M}(r_i)$ and $\mathcal{M}(r_j)$, there are two index tiles t_1 and t_2 (t_1 might be equal to t_2) such that (i) t_1 and $\mathcal{M}(r_i)$ have a common interior point, (ii) t_2 and $\mathcal{M}(r_j)$ have a common interior point, and (iii) $t_1 \cap t_2 \cap \mathcal{M}(r_i) \cap \mathcal{M}(r_j) \neq \varnothing$. This means that r_1 and r_2 are associated with t_1 and t_2, respectively, and t_1 and t_2 are not $\mathbf{DC}$, which is the only clustering relation of RCC8. For the R-tree that is built by the second strategy, together with the above assumption, we know that there must be two leaf index tiles t_1 and t_2 (t_1 might be equal to t_2) such that r_1 and r_2 are associated with t_1 and t_2, respectively, and $\mathcal{M}(r_i) \subseteq t_1$ and $\mathcal{M}(r_j) \subseteq t_2$. Therefore $t_1 \cap t_2 \supseteq \mathcal{M}(r_i) \cap \mathcal{M}(r_j) \neq \varnothing$, i.e. t_1 and t_2 are not in clustering relation $\mathbf{DC}$.

From the above discussion, we know the RCC8 relation between r_i and r_j will be calculated and stored by the R-tree clustering index. □

Proposition 10 ((**Proposition 7**)). *Given a set of possibly disconnected regions $\mathcal{D} = \{r_1, ..., r_n\}$, for CDC the qualified size of $\mathcal{D}$ for the MBR-based approach is at most as large as the qualified sizes for either the grid clustering index or the R-tree clustering index.*

Proof. Suppose the CDC relation from region $r_i \in \mathcal{D}$ to region $r_j \in \mathcal{D}$ is calculated and stored by the MBR-based approach. This implies that their MBRs have a common interior point, i.e. $\mathcal{M}^\circ(r_i) \cap \mathcal{M}^\circ(r_j) \neq \varnothing$, where $\mathcal{M}^\circ(r)$ denotes the interior of an MBR $\mathcal{M}(r)$. Note that we have $\bigcup_{k=1}^{n} \mathcal{M}(r_k) \subseteq \bigcup_{l \in J} t_l$.

For the grid clustering index, by the above assumption, we know there must be one index tile t_0 s.t. $t_0 \cap \mathcal{M}^\circ(r_i) \cap \mathcal{M}^\circ(r_j) \neq \varnothing$. By the grid clustering indexing strategy of building a clustering structure for CDC we know r_i and r_j are both associated with t_0. Therefore, the CDC relations between r_i and r_j will be calculated and stored by the grid clustering index.

For the R-tree clustering index, we only need to prove that there exist two leaf level index tiles t_1, t_2 such that $t_1 \cap t_2 \neq \varnothing$ and r_i and r_j are associated with t_1 and t_2 respectively.

We consider the two strategies of building the R-tree index separately. For the first strategy, like the above discussion for the grid clustering index, we know there is a leaf index tile t_0 such that r_i and r_j are both associated with t_0. For the second strategy, we know that there exist two leaf index tiles t_1 and t_2 (t_1 might be equal to t_2) such that r_1 and r_2 are associated with t_1 and t_2, respectively, and $\mathcal{M}(r_i) \subseteq t_1$ and $\mathcal{M}(r_j) \subseteq t_2$. Therefore $t_1^\circ \cap t_2^\circ \supseteq \mathcal{M}^\circ(r_i) \cap \mathcal{M}^\circ(r_j) \neq \varnothing$, i.e. t_1 and t_2 have a common interior point. Thus t_1 and t_2 are not in any CDC clustering relation.

From the above discussion, the CDC relation from r_i to r_j will be calculated and stored by the R-tree clustering index. □