An Efficient Representation of General Qualitative Spatial Information Using Bintrees

Leif Harald Karlsen[1] and Martin Giese[2]

1 Department for Informatics, University of Oslo, Oslo, Norway
leifhka@ifi.uio.no

2 Department for Informatics, University of Oslo, Oslo, Norway
martingi@ifi.uio.no

Abstract

In this paper we extend previous work on using bintrees as an efficient representation for qualitative information about spatial objects. Our approach represents each spatial object as a bintree satisfying the exact same qualitative relationships to other bintree representations as the corresponding spatial objects. We prove that such correct bintrees always exists and that they can be constructed as a sum of local representations, allowing a practically efficient construction. Our representation is both efficient, w.r.t. storage space and query time, and can represent many well-known qualitative relations, such as the relations in the Region Connection Calculus and Allen's Interval Algebra.

1998 ACM Subject Classification I.2.4 Knowledge Representation Formalisms and Methods

Keywords and phrases Qualitative spatial data, Bintree, Data structure

Digital Object Identifier 10.4230/LIPIcs.COSIT.2017.4

1 Introduction

Spatial and temporal data types are ubiquitous in today's software, with a growing number of spatially aware devices gathering and publishing data. Spatial and temporal data are used in a great number of highly valuable applications, such as route planning, automatic navigation, and modeling of physical processes. However, temporal and especially spatial data are normally represented as complex numerical objects, where relationships between objects are implicit, and advanced algorithms (e.g. from computational geometry) are needed to determine them. Indexing these objects for efficient query answering is also complex. During the last decades, several spatial and temporal database systems have been developed, featuring advanced indexing mechanisms and efficient numerical algorithms for answering queries over these data types (see e.g. [12, 20, 13]). Despite these advances, spatial and temporal data are still significantly more difficult to handle than more traditional types of data, often lag behind when new knowledge representations are introduced and in many cases need special treatment. The present work stems from the following observations:

1. Many applications of spatial data are mostly concerned with *qualitative* relations such as overlaps or containment of spatial objects, rather than *quantitative* properties like distance, area, etc.
2. For such qualitative applications, resorting to expensive computations on the numerical representations for each query seems wasteful. It would be sufficient to store a (pre-computed) database table for overlap, containment, or any other relations of interest, treating these relations like any other in a relational database. But this, also seems wasteful, in terms of space, since such tables could be quadratic in the number of geometries

13th International Conference on Spatial Information Theory (COSIT 2017).
Editors: Eliseo Clementini, Maureen Donnelly, May Yuan, Christian Kray, Paolo Fogliaroni, and Andrea Ballatore; Article No. 4; pp. 4:1–4:15
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

(for binary relations, cubic for ternary and so on), despite obvious redundancies, like e.g. the transitivity of the containment relation or symmetry of the overlaps relation.

3. Numerical representations of spatial objects are often subject to precision errors. E.g. even though two objects are touching in the real world, their numerical representations might not, due to insufficient precision in either their numerical representation or the measuring device reporting the objects' spatial extent. These errors are difficult or sometimes even impossible to fix numerically without introducing other errors [3]. However, if we construct a new qualitative representation of the objects, we can fix such errors during the translation by using domain knowledge about the objects. For instance, we might know that every country c touches all countries c' whenever their numerical representations have a smallest distance of 1 kilometer, or that the spatial extent of any capital of a country is contained in the extent of that country.
4. Most approaches to qualitative spatial representation can be divided into two types: they are either targeted at complex reasoning tasks (consistency checking, entailment, etc.) and is therefore not suitable as an efficient *data structure* for qualitative information extracted from a set of concrete spatial objects; or they focus on a particular set of relations for a particular type of spatial data. (See Section 7 for more details.)

Our approach is to construct a linear bintree-representation for each spatial object that are correct w.r.t. any given set of qualitative relations definable from a given first-order language. This representation scales to real-world datasets without limiting the approach to any fixed set of relations. The linear bintree [24, 25] consists of a set of bit-strings, each representing a small chunk of space obtained by recursively dividing space. Thus, bintrees represent a union of chunks of space, and two bintrees can therefore e.g. spatially overlap or one can contain the other. We can therefore make one bintree per spatial object that have the same relationships to each other as the spatial objects have, thereby becoming a *representation* of the qualitative relationships between the spatial objects.

Bintrees have the convenient property that they can be stored as a regular relation in a relational database. Furthermore, the bintrees can themselves be indexed by normal database index structures, like B-trees, since they only consist of sets of bit-strings where each bit-string can be represented by one integer. Another desirable feature of bintrees is that they allow variable resolution, so we can have low resolution (few and short bit-strings) for homogeneous areas and high resolution (many and long bit-strings) for heterogeneous areas where more detail is necessary. The bintree has previously been used as an indexing structure for geometries and as an efficient representation for images. Bintrees are now considered obsolete as index structures for geometries, as R-trees [8] and their variants (see e.g. [18] for an overview) have better performance. However, for our purpose of representing qualitative information, we will see that the bintree is a good fit.

The concrete problem this paper addresses is the following: Given a set of objects with a spatial interpretation and a set of qualitative relations, construct a bintree representation that returns the same answers to queries with the given relations over the spatial objects. We have previously constructed both theory [10] and an implementation [11] for constructing such qualitatively correct bintrees, with promising results. However, our previous work has been restricted to the construction of bintree-representations that are correct only w.r.t. part-of and overlaps relationships (as presented in Section 2). In this paper we will extend the theoretical foundation to allow for representations that are correct with respect to a more expressive set of relations.

The paper is outlined as follows: In Section 2 we introduce the spatial objects we work with and the key notions and results needed for expressing and constructing correct bintrees;

in Section 3 we explain how to construct correct bintrees and why this is a good representation for qualitative information; in the Sections 4 and 5 we extend the expressiveness of the relational language our bintrees are correct w.r.t. in two directions; in Section 6 we show several examples of common qualitative relations that our bintrees can represent; Section 7 discusses related work and Section 8 concludes the paper.

2 Spatial Objects and Correct Bintrees

We will start by introducing the central concept of spaces, the elements which we aim to represent correctly.

▶ **Definition 1.** A *space lattice* $\mathcal{S} = (S, \prec_{\mathcal{S}}, \top_{\mathcal{S}}, \bot_{\mathcal{S}})$ is a bounded, distributive lattice with top element $\top_{\mathcal{S}}$ and bottom element $\bot_{\mathcal{S}}$. We will let $\otimes_{\mathcal{S}}$ and $\oplus_{\mathcal{S}}$ be the induced meet and join operators respectively, and call the elements of S *spaces*. We will let $S^+ := S \setminus \{\bot_{\mathcal{S}}\}$.

Typical examples of such space lattices is the lattice of geometrical objects (polygons, lines points) where $\prec_{\mathcal{S}}$ is geometric containment, the lattice of temporal intervals where $\prec_{\mathcal{S}}$ is temporal containment, the lattice of sets where $\prec_{\mathcal{S}}$ is the subset-relation, and so on. Thus, the goal of this paper is to construct efficient representations of the qualitative relationships between such structures. In order to do this, we need to compute these relationships between the spatial objects, however checking all possible relationships between all possible spaces would be very complex, as this has a complexity of $\mathcal{O}(n^k)$ (for n elements with relations of arity k). A property of qualitative relationships like *overlaps* and *contains* is that they are local, that is, they depend only on the spatial parts *inside* the elements, and nothing more. Thus, we want to exploit this locality in a similar fashion as the *bucket sort*-algorithm does, where the elements to sort are first distributed into a set of of buckets/intervals partitioning the universe. The buckets are sorted individually, before being gathered into a sorted list.

In a similar fashion, we will construct a set of chunks of space partitioning the space-lattice's universe $\top_{\mathcal{S}}$, and construct locally correct representations in each chunk. We will call such a chunk a *block*. The blocks are most naturally construed in a recursive fashion where we start with $\top_{\mathcal{S}}$ and recursively split blocks into two smaller blocks, until we reach some desired property (e.g. the desired resolution or the desired number of spatial objects overlapping each block.) This splitting forms a binary tree, so each block can be represented as the path from the root ($\top_{\mathcal{S}}$) down to that block. Furthermore, such a path can be compactly represented as a bit-string (a 0-bit and 1-bit denotes a left-edge and a right-edge resp.) Note that every bit-string denotes a chunk of space, and that if s is a bit-string which is a prefix of s', then the block denoted by s spatially contains the block denoted by s'. If we let a set of bit-strings denote their union, we can represent more complex spaces that can spatially overlap and contain other spaces.

Our representation should allow efficient updates, and since relationships are locally determined, inserting a new object into our representation should only affect the representation of the blocks overlapping the object to insert. However, for such a local insert to be possible we need to know which block each representation was constructed in.

Therefore, it would seem natural to let each element's local block-representation be a set of bit-strings, each contained in that block, which satisfies the same qualitative relationships as the spatial objects they represent. A set of such bit-strings is in fact a linear bintree. The bintree is thus a binary trie data structure, similar to the quadtree and octree. For a discussion and comparison of these three structures, see e.g. [24]. Below follows the formal definition of both bit-strings and bintrees.

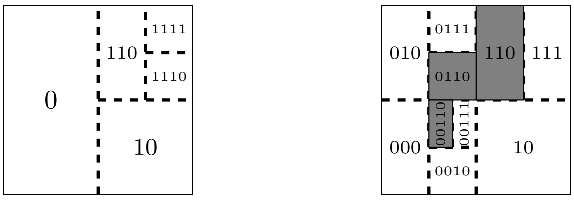

Figure 1 The left figure shows the bit-string representation of some blocks and the right figure the spatial extent (in gray) of the bintree $\{110, 0110, 00110\}$ in 2D.

▶ **Definition 2.** Let $\mathbb{B}$ to be the *set of bit-strings* with ε being the empty bit-string and $b \circ b'$ to be the *concatenation* of the bit-strings b and b'. Let the *prefix-relation* on blocks, $\lhd$, be defined as $b_1 \lhd b_2 \Leftrightarrow \exists b \in \mathbb{B}(b_2 \circ b = b_1)$ and the *neighbor-relation* on blocks, $\sim$, be defined as $b_1 \sim b_2 \Leftrightarrow \exists b \in \mathbb{B}(b_1 = b \circ 0 \wedge b_2 = b \circ 1)$. Define a *block-set* B to be a non-empty, finite set of bit-strings such that if $b \in B$ then B also contains all $b' \in \mathbb{B}$ such that either $b \sim b'$ or $b \lhd b'$.

▶ **Definition 3.** Define the *$\mathcal{T}$-lattice* $\mathcal{T} := (T, \prec_{\mathcal{T}}, \top_{\mathcal{T}}, \bot_{\mathcal{T}})$ where $T = \mathcal{P}_{fin}(\mathbb{B})$ is the set of bintrees, (where $\mathcal{P}_{fin}$ is the finite powerset) such that $t \in T$ contains no two distinct elements b_1, b_2 where either $b_1 \lhd b_2$ or $b_1 \sim b_2$. Furthermore, let $\top_{\mathcal{T}} = \{\epsilon\}$, $\bot_{\mathcal{T}} = \emptyset$, and $t \prec_{\mathcal{T}} t' \Leftrightarrow \forall b \in t \exists b' \in t(b \lhd b')$.

It should be easy to see that the $\mathcal{T}$-lattice is a space lattice. Thus, bintrees behave similarly to spaces, which allows them to be used as representations for spaces.

In Figure 1 we can see an example of both blocks and a bintree, and their spatial extent (assuming regular splitting in each space division). Note that we put no restriction on the number of dimensions our spaces has, and the same holds for our bintrees. In the same way we alternate between splitting along the x- and y-axis in the 2D case, we would cycle through all k dimensions in a k-dimensional space. We will now introduce our models, which will allow us to precisely define correctness of bintree-representations.

▶ **Definition 4.** Given a space lattice $\mathcal{S} = (S, \prec_{\mathcal{S}}, \top_{\mathcal{S}}, \bot_{\mathcal{S}})$, a finite set of constants C and a block-set B, an *S-model* $\mathcal{M}$ is a first order model over the similarity type $\langle \prec; C \cup B \rangle$ with universe S, but where $(\exists^+ z.\varphi)^{\mathcal{M}} \Leftrightarrow \varphi[s/z]^{\mathcal{M}}$ for some $s \in S^+$, and where $\epsilon^{\mathcal{M}} = \top_{\mathcal{S}}$, $\prec^{\mathcal{M}} = \prec_{\mathcal{S}}$, and $b^{\mathcal{M}} \neq \bot_{\mathcal{S}}$, $(b \circ 0)^{\mathcal{M}} \otimes_{\mathcal{S}} (b \circ 1)^{\mathcal{M}} = \bot_{\mathcal{S}}$, and $(b \circ 0)^{\mathcal{M}} \oplus_{\mathcal{S}} (b \circ 1)^{\mathcal{M}} = b^{\mathcal{M}}$ for any $b \in B$.

Note the interpretation of the new existential quantifier, and that if e.g. $\exists^+ z(z \prec c_1 \wedge z \prec c_2)$ holds in some model, then there is a non-empty intersection between c_1 and c_2 in that model. Observe also that given a space-lattice $\mathcal{S}$, the only difference between two $\mathcal{S}$-models is their interpretation of the constants $C \cup B$. The constants C will be the elements which have a spatial interpretation that we wish to correctly represent as bintrees. The constants of B will function as the *buckets* as described above. However, before we can talk about correct representations, we need to define the scope of this correctness. Our notion of correctness will be restricted to a language of first order sentences that nicely captures a core of qualitative relations, namely overlaps and containment relationships. We will in the later sections of this paper extend the expressiveness of the language.

▶ **Definition 5.** Let an *atomic spatial formula* be a first order formula on one of the two forms: $x_1 \prec x_2$ or $\exists^+ z \left(\bigwedge_{i \in I} z \prec x_i\right)$. A *spatial formula* is a first order formula $\varphi(\vec{x})$ defined by the BNF $\varphi := \psi \mid \neg\psi \mid \varphi_1 \wedge \varphi_2$, where ψ is an atomic spatial formula.

Given a set of constants C and a block-set B, an *(atomic) spatial sentence* $\varphi(\vec{c})$ is a first order sentence such that $\varphi(\vec{x})$ is an (atomic) spatial formula and $\vec{c} \in (B \cup C)^{|\vec{c}|}$.

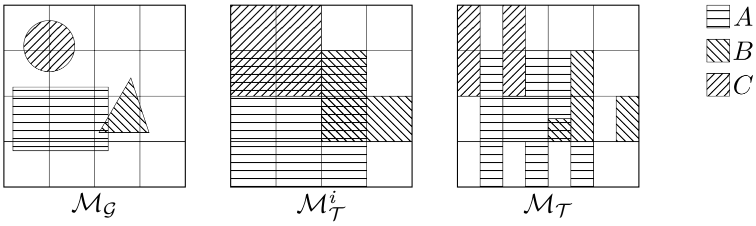

Figure 2 A figure with three polygons to the left, an example of a Γ-incorrect bintree-representation in the middle, and an example of a Γ-correct bintree-representation to the right.

▶ **Definition 6.** Given a set of spatial formulae Γ, we will say that a $\mathcal{T}$-model $\mathcal{M}_\mathcal{T}$ is Γ-*correct* w.r.t. an $\mathcal{S}$-model $\mathcal{M}_\mathcal{S}$ if $\mathcal{M}_\mathcal{T} \vDash \varphi(\vec{c}) \Leftrightarrow \mathcal{M}_\mathcal{S} \vDash \varphi(\vec{c})$ where $\vec{c} \in (B \cup C)^{|\vec{c}|}$ and $\varphi(\vec{x}) \in \Gamma$.

▶ **Example 7.** Let $\Gamma := \{x \prec y, \exists^+ z\,(z \prec x_1 \land z \prec x_2)\ \}$. In Figure 2 we can see an example of a Γ-incorrect and a Γ-correct bintree model for the constants $\{A, B, C\} \cup \{b \in \mathbb{B} \mid |b| \leq 4\}$ (where $|b|$ is the length of the bit-string b), w.r.t. a geometric model $\mathcal{M}_\mathcal{G}$. $\mathcal{M}^i_\mathcal{T}$ is just an approximation from above, which is how bintrees are normally used as index structures. We can see that such a representation is complete w.r.t. Γ, i.e. $\mathcal{M}_\mathcal{G} \vDash \varphi(\vec{c}) \Rightarrow \mathcal{M}^i_\mathcal{T} \vDash \varphi(\vec{c})$ for any spatial formula $\varphi(\vec{x}) \in \Gamma$ and any $\vec{c} \in (C \cup B)^{|\vec{c}|}$, but it is not sound, i.e. the converse implication does not necessarily hold. For instance, $\mathcal{M}^i_\mathcal{T} \vDash ov(A, C)$ but $\mathcal{M}_\mathcal{G} \nvDash ov(A, C)$, and $\mathcal{M}^i_\mathcal{T} \vDash \mathit{1011} \prec B$ but $\mathcal{M}_\mathcal{G} \nvDash \mathit{1011} \prec B$. However, $\mathcal{M}_\mathcal{T}$ is an example of a correct model, and it is easy to check that *any* spatial sentence is true in $\mathcal{M}_\mathcal{T}$ if and only if it is true in $\mathcal{M}_\mathcal{G}$.

As stated above, for efficiency reasons we will construct our bintrees locally. Thus, we need a notion of local correctness, that is, what a locally correct bintree-model is.

▶ **Definition 8.** Let $\vDash_b$ for a bit-string b, be equivalent to $\vDash$, but where $\mathcal{M}_\mathcal{S} \vDash_b c \prec d \Leftrightarrow \left(b^{\mathcal{M}_\mathcal{S}} \otimes_\mathcal{S} c^{\mathcal{M}_\mathcal{S}}\ \right) \prec_\mathcal{S} \left(b^{\mathcal{M}_\mathcal{S}} \otimes_\mathcal{S} d^{\mathcal{M}_\mathcal{S}}\ \right)$ and $\mathcal{M}_\mathcal{S} \vDash_b \exists^+ z.\varphi \Leftrightarrow \mathcal{M}_\mathcal{S} \vDash_b \varphi[s/z]$ for some $s \in S^+$ and $s \prec_\mathcal{S} b^{\mathcal{M}_\mathcal{S}}$. Given a block-set B, we will call $\mathcal{M}_\mathcal{T}$ *locally* Γ-*correct* if $\mathcal{M}_\mathcal{T} \vDash_b \varphi(\vec{c}) \Leftrightarrow \mathcal{M}_\mathcal{S} \vDash_b \varphi(\vec{c})$ for all spatial sentences $\varphi(\vec{c})$ where $\varphi(\vec{x}) \in \Gamma$ and all $\lhd$-smallest elements b of B.

So a locally correct model is a model that is correct if we limit out vision to one block at the time. We will now show that our qualitative relations are locally determined, that is, locally correct models are also globally correct.

▶ **Theorem 9.** *Given a set of constants C and a block-set B, any locally* Γ*-correct $\mathcal{T}$-model $\mathcal{M}_\mathcal{T}$ is* Γ*-correct, w.r.t. an $\mathcal{S}$-model $\mathcal{M}_\mathcal{S}$.*

Proof. Let β be set of $\lhd$-smallest elements of B. It is sufficient to prove that for any $\mathcal{S}$-model $\mathcal{M}_\mathcal{S}$ we have $\mathcal{M}_\mathcal{S} \vDash c_1 \prec c_2 \Leftrightarrow \forall b \in \beta\,(\mathcal{M}_\mathcal{S} \vDash_b c_1 \prec c_2\ )$ and $\mathcal{M}_\mathcal{S} \vDash \exists^+ z \left(\bigwedge_{i \leq k} z \prec c_i\ \right) \Leftrightarrow \exists b \in \beta \left(\mathcal{M}_\mathcal{S} \vDash_b \exists^+ z \left(\bigwedge_{i \leq k} z \prec c_i\ \right)\ \right)$ for any $c_1, \ldots, c_k \in C \cup B$. By definition the $\mathcal{M}_\mathcal{S}$-interpretation of the elements of β forms a partition on $\top_\mathcal{S}$, so $\top_\mathcal{S} = \bigoplus_{b \in \beta} b^{\mathcal{M}_\mathcal{S}}$ and $b_1^\mathcal{M} \otimes_\mathcal{S} b_2^\mathcal{M} = \bot_\mathcal{S}$. This, together with distributivity, we know that $c_1^{\mathcal{M}_\mathcal{S}} \prec_\mathcal{S} c_2^{\mathcal{M}_\mathcal{S}}$ is equivalent to $\forall b \in \beta \left(\left(b^{\mathcal{M}_\mathcal{S}} \otimes_\mathcal{S} c_1^{\mathcal{M}_\mathcal{S}}\right) \prec_\mathcal{S} \left(b^{\mathcal{M}_\mathcal{S}} \otimes_\mathcal{S} c_2^{\mathcal{M}_\mathcal{S}}\right)\ \right)$ for any $c_1, c_2 \in C \cup B$. By similar arguments, we have that $\exists z \in S^+ \left(\bigwedge_{i \leq k} z \prec_\mathcal{S} c_i^{\mathcal{M}_\mathcal{S}}\right)$ is equivalent to the local $\exists b \in \beta \exists z \in S^+ \left(\bigwedge_{i \leq k} z \prec \left(b^{\mathcal{M}_\mathcal{S}} \otimes_\mathcal{S}\ c_i^{\mathcal{M}_\mathcal{S}}\right)\right)$ for any $c_1, \ldots, c_k \in C \cup B$. ◀

▶ **Theorem 10.** *For any $\mathcal{S}$-model $\mathcal{M}_\mathcal{S}$ there exists a locally* Γ*-correct bintree-model $\mathcal{M}_\mathcal{T}$.*

Proof. Proof done by model construction: For each $b \in \beta$, construct the set of all locally true atomic spatial sentences occurring (either positively or negatively) in some $\varphi \in \Gamma$:

$$T_b := \left\{ \psi(\vec{c}) \mid \mathcal{M}_{\mathcal{S}} \vDash_b \psi(\vec{c}), \bigwedge_{1 \leq i \leq |\vec{c}|} \left(c_i \in C \cup B \land c_i^{\mathcal{M}_{\mathcal{S}}} \otimes b^{\mathcal{M}_{\mathcal{S}}} \neq \bot_{\mathcal{S}} \right) \right\}.$$

Any $\mathcal{T}$-model of all T_bs is locally Γ-correct. Then, let T_b' be the skolemization of T_b, and T_b'' be the set of atoms occurring in any sentence in T_b'. Define K_b^+ to be the set of all constants occurring in T_b'', and $K_b^\perp := (C \cup B) \setminus K_b^+$. So $\{K_b^+, K_b^\perp\}$ partitions $C \cup B$ and K_b^+ is the set of constants that should have a non-empty interpretation, locally in b.

We will now construct the $\mathcal{T}$-model. First, for each $b \in \beta$, generate a set $W_b \subseteq \mathbb{B}$ of size $|K_b^+|$ of pairwise $\lhd$-unrelated bit-strings b' such that $b' \lhd b$. Then, let $w_b : \mathcal{K}_b^+ \to W_b \cup \{b\}$ be a bijective function on $\mathcal{K}_b^+ \setminus B$ and $w_b(b') = b$ for $b' \in \mathcal{K}_b^+ \cap B$. Then, define $I_b(c) := \bigoplus_{\mathcal{T}} \{\{w_b(c')\} \mid (c' \prec c) \in T_b'', c' \in K_b^+\}$ for each $c \in \mathcal{K}_b^+$, and $I_b(c) := \bot_{\mathcal{T}}$ for $c \in \mathcal{K}_b^\perp$. So I_b is the locally correct interpretation of the constants in $C \cup B$, and it should be clear that $\left(\left(c_1^{\mathcal{M}_{\mathcal{S}}} \otimes_{\mathcal{S}} b^{\mathcal{M}_{\mathcal{S}}} \right) \prec_{\mathcal{S}} \left(c_2^{\mathcal{M}_{\mathcal{S}}} \otimes_{\mathcal{S}} b^{\mathcal{M}_{\mathcal{S}}} \right) \right)$ if and only if $I_b(c_1) \prec_{\mathcal{T}} I_b(c_2)$ and $\exists z \in S^+ \left(\bigwedge_{i \leq k} z \prec_{\mathcal{S}} \left(b^{\mathcal{M}_{\mathcal{S}}} \otimes_{\mathcal{S}} c_i^{\mathcal{M}_{\mathcal{S}}} \right) \right)$ if and only if $\exists z \in T^+ \left(\bigwedge_{i \leq k} z \prec_{\mathcal{T}} I_b(c_i) \right)$ for any $c_1, \ldots, c_k \in C \cup B$ and any $b \in \beta$. Finally, let $c^{\mathcal{M}_{\mathcal{T}}} := \bigoplus_{\mathcal{T}} \{I_b(c) \mid b \in \beta\}$ for each $c \in C \cup B$. $\mathcal{M}_{\mathcal{T}}$ is now a Γ-correct $\mathcal{T}$-model w.r.t $\mathcal{M}_{\mathcal{S}}$. ◀

3 How To and Why Construct Correct Bintrees

The proof of Theorem 10 illustrates how one could design an algorithm for construction of correct bintree-models. We can write an almost direct translation of the steps in the proof to an algorithm. That is, for each $b \in \beta$ do the following: Find all $c \in C \cup B$ overlapping b and compute their Γ-relationships, T_b; skolemize and extract the atomic sentences, T_b''; generate a set of blocks and assign each non-empty element a block, and propagate according to the $\prec$-relationships in T_b''; finally, sum up the local representations to form the model. The algorithmic complexity of such a model construction is $|\beta|$ times the complexities of first constructing T_b'' and then generating and distributing the elements of W_b. It should be easy to see that the latter has complexity $\mathcal{O}(|T_b''|)$. Note that constructing T_b' from T_b and T_b'' from T_b' are both linear in the size of T_b. Lastly, we have that constructing T_b requires computing whether $\mathcal{M}_{\mathcal{S}} \vDash \varphi(\vec{c})$ holds for each atomic spatial sentence generated from the atomic spatial formulas of Γ and the constants $B \cup C$. This gives us a total complexity of $\mathcal{O}(|\beta| \cdot o^k)$, where $o = \max_{b \in beta} |\{c \in C \cup B \mid \mathcal{M}_{\mathcal{S}} \vDash ov(c, b)\}|$, that is, the largest number of elements from $C \cup B$ that overlaps any $b \in \beta$, and k is the largest number of free variables occurring in any atomic spatial formula occurring in any $\varphi \in \Gamma$. This means that we in practice *can* construct correct bintree-models for any $\mathcal{S}$-model, however, *why* still remains to be answered. Below we discuss the main properties of the representation making it suitable for representing qualitative information.

The bintrees can be stored and queried in a relational database as a binary relation $(id, block)$, where we encode the bit-strings as integers and where both the IDs and the bit-string integers can be indexed by a normal B-tree. This allows for highly efficient query answering, in the complexity class AC_0 [1], of queries of the form "given $a \in C \cup B$ and $R \in \Gamma$, find all x such that $R(a, x)$ holds" and "given $a, b \in C \cup B$ and $R \in \Gamma$, check whether $R(a, b)$ holds". In [11] we discuss this representation in more detail and present a benchmark that shows that overlaps and containment queries are on average 2.7 times faster over our correct bintrees than over the corresponding geometries. The comparison was done with

PostGIS [20], a state of the art geospatial database, over real-world datasets where the largest sets has over a million geometries.

Note that our constructions also allow a more efficient insertion than reconstructing the entire model upon each insert: Assume we already have constructed a Γ-correct model $\mathcal{M}_{\mathcal{T}}$ for the $\mathcal{S}$-model $\mathcal{M}_{\mathcal{S}}$ and constants $C \cup B$, but now want to construct a Γ-correct model for the extended model $\mathcal{M}'_{\mathcal{S}}$ for $C \cup C' \cup B$. Since we only need local Γ-correctness, we only need to update I_b for each $b \in \beta$ where $\mathcal{M}'_{\mathcal{S}} \vDash ov(c, b)$ for any $c \in C'$. Thus, a larger B gives a more efficient insert-operation as we have a higher resolution. Observe also that the only requirements we put on the interpretations of the elements of B, is that $\{b \circ 0, b \circ 1\}$ partitions b. Thus, we are free to interpret $b \circ 0$ and $b \circ 1$ in such a way that there is approximately the same number of elements from C that overlap each. This will evenly spread the elements of C over the elements of β, thus making each T_b about equally complex to compute. This is important, as it can greatly reduce the value of o in the complexity measure. We present an algorithm for construction and update of Γ-correct bintrees with such balanced splitting of B in [11], with $\Gamma = \{x \prec y, \exists^+ z(\bigwedge_{i \leq k} z \prec x_i)\}$ for arbitrary k.

Our representation is also compact, as it does not need to explicitly store reflexive, symmetric or transitive closures of the containment and overlaps relationships. There are also many optimizations one can do to get an even more compact and efficient representation: E.g. we can remove all sentences φ from T_b if there is some sentence $\varphi' \in T_b$ such that $\varphi' \rightarrow \varphi$. This will remove all redundant overlaps-witnesses (either implied by a containment-relationship or another overlaps-relationship of higher arity) and reduce the overall size of the bintrees. In the benchmark in [11] we show that our bintree-representation uses only 62% of the space of the corresponding geometry-datasets, and only 22% of the explicit representations, for the largest datasets.

4 Extension: Roles

We have now seen that we can construct a correct bintree-representation for any space lattice, but the correctness is only for spatial sentences of containment and overlaps relationships. We will now see that a small extension to our bintree representations allows us to accommodate a much more interesting set of relationships. First observe that we, e.g., can express the well known RCC8-relations (see e.g. [22, 5]) with only containment and overlaps relations, if we can relate the different *types of parts*:

$$\begin{aligned} DJ(x,y) &:= \neg ov(x,y) & EC(x,y) &:= ov(x,y) \wedge \neg ov(x^\circ, y^\circ) \\ PO(x,y) &:= ov(x^\circ, y^\circ) \wedge (x \not\prec y) \wedge (y \not\prec x) & EQ(x,y) &:= x \prec y \wedge y \prec x \\ TPP(x,y) &:= x \prec y \wedge ov(\partial x, \partial y) \wedge (y \not\prec x) & NTPP(x,y) &:= x \prec y \wedge \neg ov(\partial x, \partial y) \end{aligned}$$

where $ov(x, y) := \exists^+ z(z \prec x \wedge z \prec y)$, ∂x is the boundary of x, x° is the interior of x and $x \not\prec y$ is short for $\neg(x \prec y)$. We will therefore extend our definitions above with the notion of *roles*, which allows us to talk about the different parts of a space, e.g. interior and boundary.

▶ **Definition 11.** A *role* is a set of names. A *role-set* is a set of roles containing $\emptyset$.

As we will see shortly, we only need roles that consist of a single name to express the relations of RCC8, namely i for interior and b for boundary. However, we will also see examples where using multiple names to denote a part is useful.

▶ **Definition 12.** Given a role-set R, an *R-roled space lattice* $\mathcal{S}$ is a tuple $(S, \prec_{\mathcal{S}}, \top_{\mathcal{S}}, \bot_{\mathcal{S}}, \pi_{\mathcal{S}})$ where $(S, \prec_{\mathcal{S}}, \top_{\mathcal{S}}, \bot_{\mathcal{S}})$ is a space lattice and $\pi_{\mathcal{S}} : R \times S \to S$ is a function where $\pi_{\mathcal{S}}(\emptyset, s) = s$ and $\pi_{\mathcal{S}}(r \cup u, s) = \pi_{\mathcal{S}}(r, s) \otimes_{\mathcal{S}} \pi_{\mathcal{S}}(u, s)$ for any $(r, u) \in R^2$ such that $r \cup u \in R$ and any $s \in S$.

The reader can read $\pi_{\mathcal{S}}(r, a)$ as "a's r-part". Intuitively one can think of a $\{n_1, \ldots, n_k\}$-part as an intersection of all the $\{n_i\}$-parts. For instance the role $\{i, h\}$, where h is short for hole, denotes holes in an interior, whereas $\{i\}$ denotes all of the interior, both with and without holes. Observe also that we always have $\pi_{\mathcal{S}}(r, s) \prec_{\mathcal{S}} s$. We will now introduce the corresponding bintrees.

▶ **Definition 13.** Given a role-set R, an *R-roled block* is a pair (r, b) such that $b \in \mathbb{B}$ and $r \in R$. Let $\mathbb{B}_R$ be the set of R-roled blocks. Also let $\delta(r, t) := \{b \mid (r, b) \in t\}$ for any $r \in R$ and $t \subseteq \mathbb{B}_R$ and let $\Sigma_R(t) := \bigoplus_{r \in R} \delta(r, t)$.

An *R-roled bintree* t is an element of $\mathcal{P}_{fin}(\mathbb{B}_R)$, such that for any role $r \in R$ we have that $\delta(r, t)$ is a bintree, and $\delta(r, t) \otimes_{\mathcal{T}} \delta(u, t) = \bot_{\mathcal{T}}$ for any $r \neq u$. Let T_R be the set of R-roled bintrees. Furthermore, let $\top_{\mathcal{T}_R} := \{(\emptyset, \varepsilon)\}$, $\bot_{\mathcal{T}_R} := \emptyset$, $\pi_{\mathcal{T}_R}(r, t) := \{(u, b) \in t \mid r \subseteq u\}$, $t \prec_{\mathcal{T}_R} t' \Leftrightarrow \Sigma_R(t) \prec_{\mathcal{T}} \Sigma_R(t')$ and $\mathcal{T}_R := (T_R, \prec_{\mathcal{T}_R}, \top_{\mathcal{T}_R}, \bot_{\mathcal{T}_R}, \pi_{\mathcal{T}_R})$.

While the different roles for the parts are implicitly defined for spaces like geometries, (such as being the interior of a polygon), we explicitly state the roles each block should have in the bintree. So the boundary of a bintree t, $\pi_{\mathcal{T}}(\{b\}, t)$, is the set of blocks having a role r such that $b \in r$. We can then define the touching relation as $ov(\pi_{\{b\}}(x), \pi_{\{b\}}(y))$. So even though two bintrees seem to touch geometrically (e.g. if one has a block b and the other a block b' and $b \sim b'$) they will not necessarily touch according to our definition. This makes it easier for us to construct correct bintree-models, as we still only have to care about overlaps and part-of relationships. Note also that it is possible to construct bintree-models that satisfy sentences that are unsatisfiable by any $\mathcal{S}$-model for a particular space lattice $\mathcal{S}$. For instance, it is easy to make a bintree model with two objects that have a partially overlapping interior, but that have disjoint boundaries, which is impossible for any geometrical model. Thus, we cannot use our representation for reasoning (that is, make a representation for a set of sentences and then query for all entailments). However, as our bintrees only function as a representation of the relationships of a given $\mathcal{S}$-model and is constructed to satisfy exactly these, this is not a problem.

▶ **Definition 14.** Given an R-roled space lattice $\mathcal{S}$, a set of constants C, and a block-set B, an *R-roled $\mathcal{S}$-model* $\mathcal{M}$ is a first order model over the similarity type $\langle \prec; \pi; C \cup B \rangle$, where π is a family of unary function symbols π_r for each $r \in R$, that is an $\mathcal{S}$-model over $\langle \prec; C \cup B \rangle$ and where $\pi_r(c)^{\mathcal{M}_{\mathcal{S}}} = \pi_{\mathcal{S}}(r, c^{\mathcal{M}_{\mathcal{S}}})$ for any $r \in R$ and $c \in C \cup B$.

▶ **Definition 15.** Given a role-set R, an *R-roled atomic spatial formula* is a first order formula on one of the forms $\pi_{r_1}(x) \prec \pi_{r_2}(y)$ or $\exists^+ z \left(\bigwedge_{i \leq k} z \prec \pi_{r_i}(x_i) \right)$ for some $r_1, \ldots, r_k \in R$. Let R-roled formulae and R-roled (atomic) spatial sentences be defined analogously as in Definition 5, but where ψ is an R-roled (atomic) spatial formula.

Note that $(\pi_\emptyset(s))^{\mathcal{M}_{\mathcal{S}}} = s$ for any R-roled $\mathcal{S}$-model $\mathcal{M}_{\mathcal{S}}$, so we sometimes write x instead of $\pi_\emptyset(x)$ in the definitions of spatial formulae. To save ink, let $\bar{r} = \{r\}$ for any role-name r.

▶ **Example 16** (RCC8). Assume we have the names b for "boundary", and i for "interior", where $\pi_{\bar{b}}(x)$ denotes x's boundary and $\pi_{\bar{i}}(x)$ denotes x's interior, we can now express the RCC8-relations with Γ equal to the set of formulae:

$$
\begin{aligned}
DJ(x, y) &:= \neg ov(x, y) & EC(x, y) &:= ov(x, y) \wedge \neg ov(\pi_{\bar{i}}(x), \pi_{\bar{i}}(y)) \\
PO(x, y) &:= ov(\pi_{\bar{i}}(x), \pi_{\bar{i}}(y)) \wedge (x \not\prec y) \wedge (y \not\prec x) & EQ(x, y) &:= x \prec y \wedge y \prec x \\
TPP(x, y) &:= x \prec y \wedge ov(\pi_{\bar{b}}(x), \pi_{\bar{b}}(y)) \wedge (y \not\prec x) & NTPP(x, y) &:= x \prec y \wedge \neg ov(\pi_{\bar{b}}(x), \pi_{\bar{b}}(y))
\end{aligned}
$$

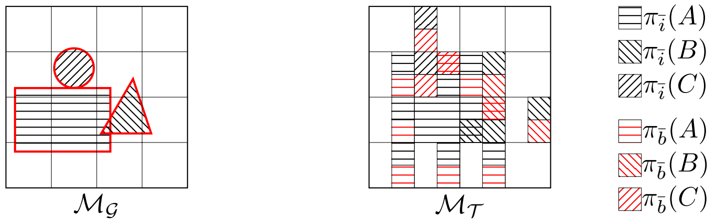

Figure 3 Polygons and an RCC8-correct bintree-model.

Let G be set of two-dimensional geometries (i.e. polygons, line-strings, points) contained in some universe $\top_{\mathcal{G}}$ and with $\prec_{\mathcal{G}}$ being geometric containment, then $\mathcal{G}$ is a space lattice. So a Γ-correct $\mathcal{T}_R$-model w.r.t. $\mathcal{G}$-models $\mathcal{M}$ will correctly represent all RCC8-relations between the elements of C as interpreted by $\mathcal{M}$. In Figure 3 we see an example of a correct bintree-model with respect to the RCC8-relations.

▶ **Theorem 17.** *Any R-roled $\mathcal{T}_R$-model $\mathcal{M}_{\mathcal{T}}$ is Γ-correct if and only if it is locally Γ-correct, w.r.t. an R-roled $\mathcal{S}$-model $\mathcal{M}_{\mathcal{S}}$. Furthermore, for any R-roled $\mathcal{S}$-model $\mathcal{M}_{\mathcal{S}}$ there exists a locally Γ-correct R-roled $\mathcal{T}_R$-model $\mathcal{M}_{\mathcal{T}}$.*

Proof. The arguments for the first part are analogous to the proof of Theorem 9, just substitute c_i with $\pi_{r_i}(c_i)$.

The second part is done by a similar model construction as for the proof of Theorem 10. So, construct T_b'' in the same way, but note that now the elements of T_b'' are on the forms $\pi_r(c_1) \prec \pi_u(c_2)$ and $v \prec \pi_r(c)$. Now, define K_b^+ to be the set of expressions (v and $\pi_r(c)$ where v is a skolem-constant and $c \in C \cup B$) occurring in T_b'', and $K_b^\perp$ as before. Let W_b be a set of size $|K_b^+|$ of pairwise $\lhd$-unrelated bit-strings b' where $b' \lhd b$. Then, let $w_b : K_b^+ \to W_b \cup \{b\}$ be a function such that assigns a unique element from W_b to each $e \in K_b^+ \setminus \{\pi_\emptyset(b') \mid b' \in B\}$, and $w_b(e) := b$ for each $e \in K_b^+ \cap \{\pi_\emptyset(b') \mid b' \in B\}$. We then define $I_b(\pi_r(c)) := \bigoplus_{\mathcal{T}_R} \{\{(r, w_b(e))\} \mid (e \prec \pi_r(c)) \in T_b'', e \in K_b^+\}$ for each $\pi_r(c) \in \mathcal{K}_b^+$ and $I_b(e) := \perp_{\mathcal{T}_R}$ for $e \in \mathcal{K}_b^\perp$. It should now be clear that $\left(b^{\mathcal{M}_{\mathcal{S}}} \otimes_{\mathcal{S}} (\pi_r(c_1))^{\mathcal{M}_{\mathcal{S}}}\right) \prec_{\mathcal{S}} \left(b^{\mathcal{M}_{\mathcal{S}}} \otimes_{\mathcal{S}} (\pi_u(c_2))^{\mathcal{M}_{\mathcal{S}}}\right)$ if and only if $I_b(\pi_r(c_1)) \prec_{\mathcal{T}_R} I_b(\pi_u(c_2))$ and $\exists z \in S^+ \left(\bigwedge_{i \le k} z \prec_{\mathcal{S}} \left(b^{\mathcal{M}_{\mathcal{S}}} \otimes_{\mathcal{S}} (\pi_{r_i}(c_i))^{\mathcal{M}_{\mathcal{S}}}\right)\right)$ if and only if $\exists z \in T_R^+ \left(\bigwedge_{i \le k} z \prec_{\mathcal{T}_R} I_b(\pi_{r_i}(c_i))\right)$ for any $c_1, \ldots, c_k \in C \cup B$ and any $b \in \beta$. Finally, we let $c^{\mathcal{M}_{\mathcal{T}}} := \bigoplus_{b \in \beta} \bigoplus_{r \in R} I_b(\pi_r(c))$ for each $c \in C \cup B$. $\mathcal{M}_{\mathcal{T}}$ is now an R-roled $\mathcal{T}_R$-model that satisfies exactly the true spatial sentences of $\mathcal{M}_{\mathcal{S}}$ generated from Γ and $C \cup B$. ◀

From the above proof, we can see that the construction of correct roled bintrees is done in a similar fashion as the normal bintrees, and we only need a minor update of any algorithm used for constructing normal correct bintrees.

Observe also that we can compress our roled bintrees in the following manner: Assume that in the set of skolemized atoms T_b'' we have a $\pi_r(c)$ such that $(\pi_r(c) \prec e_1) \in T_b'' \Leftrightarrow (\pi_u(c) \prec e_1) \in T_b''$ and $(e_2 \prec \pi_r(c)) \in T_b'' \Leftrightarrow (e_2 \prec \pi_u(c)) \in T_b''$ for any expressions $e_1, e_2 \in K_b^+$. If then there is no formula $\varphi(\vec{x}) \in \Gamma$ such that $eq(\pi_r(c), \pi_u(c)) \Leftrightarrow \varphi(\vec{c})$, we can let $w_b(r, c) = w_b(u, c)$, thus reducing the size of our bintree-representation. This can for instance be done for the RCC8-relations (letting $eq(\pi_{\bar{b}}(c), \pi_{\bar{i}}(c))$ cannot introduce a new relationship, if $\pi_{\bar{b}}(c)$ and $\pi_{\bar{i}}(c)$ has the exact same relationships to other elements).

Note also that any role can be represented as a fixed length bit-string by enumerating all role-names occurring in Γ and represent each role r as the bit-string having 1s at the bit-positions corresponding to the numbers given to the role-names in r, and 0 everywhere else. We can then represent our roled bintrees as ternary relations $(id, block, role)$ where each column can be index by a normal B-tree. Thus, querying roled bintrees is almost as efficient as querying our normal bintrees, as we only need to consult one additional index-structure (the B-tree over the *role*-column) during query execution.

5 Extension: Order

Introducing roles allows us to construct much richer bintree-models. However, having only the part-of relations allows only relations based on sharing of different types of parts, we are still unable to describe many interesting qualitative relationships, such as temporal relationships, relative size and relative direction. In this section we will extend our language to also include a different type of partial order which will enable us to express these relationships.

▶ **Definition 18.** An *ordered R-roled space lattice* $\mathcal{S}$ is a tuple $(S, \prec_\mathcal{S}, <_\mathcal{S}, \top_\mathcal{S}, \bot_\mathcal{S}, \pi_\mathcal{S})$ where $(S, \prec_\mathcal{S}, \top_\mathcal{S}, \bot_\mathcal{S}, \pi_\mathcal{S})$ is an R-roled space lattice and $<_\mathcal{S}$ is a strict partial order such that if $a <_\mathcal{S} b$ then $a \otimes_\mathcal{S} b = \bot_\mathcal{S}$ and for any pair $c, d \in S$ we have $c \prec_\mathcal{S} a \wedge d \prec_\mathcal{S} b \ \rightarrow c <_\mathcal{S} d$.

The reader can read the statement $x < y$ as "x is *before* y". The rest of the definitions are analogous to before:

▶ **Definition 19.** Let $t <_\mathcal{T} t' \Leftrightarrow \forall b \in t \forall b' \in t'(b <_\mathbb{B} b')$ where $b <_\mathbb{B} b'$ for bit-strings b, b' iff there exists some b'' such that $b \lhd b'' \circ 0$ and $b' \lhd b'' \circ 1$. Then let $t <_{\mathcal{T}_R} t' \Leftrightarrow \Sigma_R(t) <_\mathcal{T} \Sigma_R(t')$ and $\mathcal{T}_R^< := (T_R, \prec_{\mathcal{T}_R}, <_{\mathcal{T}_R}, \top_{\mathcal{T}_R}, \bot_{\mathcal{T}_R}, \pi_{\mathcal{T}_R})$.

▶ **Definition 20.** Given an ordered R-roled space lattice $\mathcal{S}$, a set of constants C, and a block-set B, an *ordered R-roled $\mathcal{S}$-model* $\mathcal{M}$ is a first order model over the similarity type $\langle \prec, <; \pi; C \cup B \rangle$ that is an R-roled $\mathcal{S}$-model over $\langle \prec; \pi; C \cup B \rangle$, and where $(<)^\mathcal{M} = <_\mathcal{S}$ and $(b \circ 0)^{\mathcal{M}_\mathcal{S}} <_\mathcal{S} (b \circ 1)^{\mathcal{M}_\mathcal{S}}$ for any $(b \circ 0), (b \circ 1) \in B$.

▶ **Definition 21.** Let an *atomic ordered R-roled spatial formula* be a first order formula that is either an atomic R-roled spatial formula or a formula on the form $x < y$. Let *ordered R-roled spatial formulae* and *(atomic) ordered R-roled spatial sentences* be defined analogously as R-roled spatial formulae and (atomic) R-roled spatial sentences, but where each ψ is an atomic ordered R-roled spatial formula.

▶ **Example 22** (Allen's Interval Algebra)**.** Assume we have the role-names i for *interior*, f for *first*, l for *last*, and the role-set $R := \{\emptyset, \bar{i}, \bar{f}, \bar{l}\}$. Let $\pi_{\bar{f}}(x)$ denote the interval consisting of only x's first point, and $\pi_{\bar{l}}(x)$ denote the interval consisting of only x's last point, and $\pi_{\bar{i}}(x)$ is the interior of x's interval. We can then express the relations of Allen's Interval Algebra [2]:

$$\begin{aligned} &\mathit{before}(x,y) := \pi_{\bar{l}}(x) < \pi_{\bar{f}}(y) && \mathit{meets}(x,y) := \mathit{eq}(\pi_{\bar{l}}(x), \pi_{\bar{f}}(y)) \\ &\mathit{overlaps}(x,y) := \mathit{ov}(\pi_{\bar{i}}(x), \pi_{\bar{i}}(y)) \wedge (x \not\prec y) \wedge (y \not\prec x) && \mathit{equal}(x,y) := \mathit{eq}(x,y) \\ &\mathit{starts}(x,y) := \mathit{eq}(\pi_{\bar{f}}(x), \pi_{\bar{i}}(y)) \wedge \pi_{\bar{l}}(x) \prec \pi_{\bar{i}}(y) && \mathit{during}(x,y) := \pi_{\bar{f}}(x) \prec \pi_{\bar{i}}(y) \wedge \pi_{\bar{l}}(x) \prec \pi_{\bar{i}}(y) \\ &\mathit{ends}(x,y) := \mathit{eq}(\pi_{\bar{l}}(x), \pi_{\bar{l}}(y)) \wedge \pi_{\bar{f}}(x) \prec \pi_{\bar{i}}(y) && \mathit{after}(x,y) := \pi_{\bar{l}}(y) < \pi_{\bar{f}}(x) \end{aligned}$$

Given the set I of time intervals contained in some universe $\top_\mathcal{I}$, with $\prec_\mathcal{I}$ being temporal containment, and $x <_\mathcal{I} y$ is the temporal *before*, it should be obvious that this forms an ordered R-roled space lattice. Thus, any correct $\mathcal{T}_R^<$-model w.r.t. such an $\mathcal{I}$-model $\mathcal{M}$ will correctly represent all Allen's Interval-relations between the elements of $C \cup B$ as $\mathcal{M}$.

▶ **Theorem 23.** *Any ordered R-roled $\mathcal{T}_R^<$-model $\mathcal{M}_\mathcal{T}$ is Γ-correct if and only if it is locally Γ-correct, w.r.t. an ordered R-roled $\mathcal{S}$-model $\mathcal{M}_\mathcal{S}$. Furthermore, for any ordered R-roled $\mathcal{S}$-model $\mathcal{M}_\mathcal{S}$ there exists a locally Γ-correct ordered R-roled $\mathcal{T}_R^<$-model $\mathcal{M}_\mathcal{T}$.*

Proof. For the first part, note that, since β is a partition of $\top_{\mathcal{S}}$ and $(b \circ 0) <_{\mathcal{S}} (b \circ 1)$ for any $(b \circ 0), (b \circ 1) \in B$, we have that $<_{\mathcal{S}}$ is a total order on β. This implies that $e_1^{\mathcal{M}_{\mathcal{S}}} <_{\mathcal{S}} e_2^{\mathcal{M}_{\mathcal{S}}}$ if and only if $\forall b \in \beta \left(\left(e_1^{\mathcal{M}_{\mathcal{S}}} \otimes_{\mathcal{S}} b^{\mathcal{M}_{\mathcal{S}}} \right) <_{\mathcal{S}} \left(e_2^{\mathcal{M}_{\mathcal{S}}} \otimes_{\mathcal{S}} b^{\mathcal{M}_{\mathcal{S}}} \right) \right)$. The rest of the proof is analogous to the proof of Theorem 9.

For the second part, we again have to construct a locally correct model. So, construct T_b'' in the same way as before for each $b \in \beta$, but this time the elements of T_b'' can also be on the form $e_1 < e_2$. Let W_b be as before but now with size $2|K_b^+|$. We then let $c <_b d \Leftrightarrow (c < d) \in T_b''$, and $<_b^t$ be some strict total ordering on K_b^+ containing $<_b$. Now, define $w_b(c) := \{b_c^f\} \oplus \{b_c^l\}$ (intuitively, one can think of b_c^f and b_c^l as representing the $<$-first and last part of c, respectively) for some $b_c^f, b_c^l \in W_b$ such that $b_c^f <_{\mathbb{B}} b_c^l$ and $c <_b d \Rightarrow b_c^l <_{\mathbb{B}} b_d^f$ and $c \not<_b d \wedge c <_b^t d \Rightarrow (b_c^f <_{\mathbb{B}} b_d^f <_{\mathbb{B}} b_c^l <_{\mathbb{B}} b_d^l)$. Now $w_b(c)$ and $w_b(d)$ are disjoint and $c <_b d \Leftrightarrow w_b(c) <_{\mathcal{T}_R} w_b(d)$ for any pair of distinct $c, d \in K_b^+$. We then define $I_b(e)$ and $\mathcal{M}_{\mathcal{T}}$ in the same way as before. Now, $\mathcal{M}_{\mathcal{T}}$ is an ordered R-roled model satisfying exactly the same ordered R-roled sentences generated from Γ as $\mathcal{M}_{\mathcal{S}}$. ◀

Again we see that the construction of correct bintrees with order requires only a small extension to the previous algorithm. Furthermore, a nice feature of encoding bit-strings as integers as described in Section 3 is that the $<$-ordering of the blocks corresponds to the normal $<$-ordering on their integer representations, thus we can reuse the B-tree index over the blocks to efficiently answer $<$-queries as well.

6 Expressiveness and More Examples

▶ **Example 24** (Holes). To both $\mathcal{G}$ and $\mathcal{I}$ we can add an additional role-name, h, for "hole", that can be combined with e.g. i to represent holes in the interior of a polygon or interval, or with b to represent geometries that have an open boundary. We can now express:

$$\begin{aligned} \mathit{surroundedBy}(x,y) &:= x \prec \pi_{\{h,i\}}(y) & \mathit{hasHoles}(x) &:= \exists^+ z(z \prec \pi_{\bar{h}}(x)) \\ \mathit{hasOpenBoundary}(x) &:= \pi_{\bar{b}}(x) \prec \pi_{\{b,h\}}(x) & \mathit{hasHole}(x,y) &:= \mathit{eq}(\pi_{\{h,i\}}(x), y) \end{aligned}$$

▶ **Example 25** (Relative size and direction). One dimensional attributes like size, length, projection down to the north-south and east-west axis can easily be represented by introducing an appropriate role-name, e.g. d, and let $\pi_{\mathcal{S}}(\bar{d}, x) <_{\mathcal{S}} \pi_{\mathcal{S}}(\bar{d}, y)$ hold if x has a smaller value than y on the d-axis. If we then also let for each $b \in \beta$, $\pi_{\mathcal{S}}(\bar{d}, b)$ be an interval along this axis such that β contains both the smallest and largest values, our constructing algorithm will be a normal bucket-sort with β being the set of buckets.

If we introduce the role-names n for the projection along the north-south and e for the projection down to the east-west, we can express the following relations from the Cardinal Direction Calculus[14], e.g.:

$$\begin{aligned} \mathit{northOf}(x,y) &:= \pi_{\bar{n}}(y) < \pi_{\bar{n}}(x) \wedge \mathit{ov}(\pi_{\bar{e}}(x), \pi_{\bar{e}}(y)) \\ \mathit{northEastOf}(x,y) &:= \pi_{\bar{n}}(y) < \pi_{\bar{n}}(x) \wedge \pi_{\bar{e}}(y) < \pi_{\bar{e}}(x) \end{aligned}$$

and the rest of the directional-relations are defined similarly. Note that $\pi_{\mathcal{S}}(\bar{n}, x)$ and $\pi_{\mathcal{S}}(\bar{e}, x)$ is the projection of a two-dimensional object down to the each dimension. We can of course also do this for three-dimensional (or higher) objects and introduce a role-name, u for the up-down axis, and relations such as $\mathit{above}(x,y)$ and $\mathit{between}(x,y,z)$. If we combine the directional roles with the interior-role, e.g. $\{i, u\}$, we can express

$$\begin{aligned} \mathit{onTopOf}(x,y) := \; & \mathit{ov}(\pi_{\{i,n\}}(x), \pi_{\{i,n\}}(y)) \wedge \mathit{ov}(\pi_{\{i,e\}}(x), \pi_{\{i,e\}}(y)) \wedge \\ & \mathit{ov}(\pi_{\{b,u\}}(x), \pi_{\{b,u\}}(y)) \wedge \pi_{\{i,u\}}(y) < \pi_{\{i,u\}}(x) \end{aligned}$$

that is, x and y overlap in the two-dimensional plane, but x and y are touching along the up-down axis, yet x's interior is above y's.

▶ **Example 26** (Orientation). If we have the directional roles $\{n, e\}$ as described above, we can introduce two more role-names f for *front* and b for *back*, and then introduce orientational relations, e.g $\mathit{northOriented}(x) := \pi_{\{n,b\}}(x) < \pi_{\{n,f\}}(x) \wedge ov(\pi_{\{e,b\}}(x), \pi_{\{e,f\}}(x))$ and similarly for the rest of the directions. If we allow unions of relations in our query language (this is trivial in SQL), we can express relative orientation, that is, $\mathit{orientedTowards}(x, y)$ as the union of the 8 relations on the form $\mathit{northOf}(x, y) \wedge \mathit{southOriented}(x)$.

▶ **Example 27** (Egg-Yolk). If we have a space-lattice $\mathcal{S}$ with indeterminate boundaries (that is, an inner and outer boundary where the real boundary is somewhere in between) we can introduce two new role-names y, for *yolk*, and w, for *white*, and let $\pi_{\mathcal{S}}(\bar{y}, s)$ be the region within the inner boundary and $\pi_{\mathcal{S}}(\bar{w}, s)$ be the region within the outer boundary. We can then introduce all the 46 relations from the Egg-Yolk RCC5 calculus [6], e.g.:

$$R_2(x, y) := PO'(\pi_{\bar{w}}(x), \pi_{\bar{w}}(y)) \wedge \neg ov(\pi_{\bar{y}}(x), y) \wedge \neg ov(\pi_{\bar{y}}(y), x)$$
$$R_{11}(x, y) := PO'(\pi_{\bar{w}}(x), \pi_{\bar{w}}(y)) \wedge PO'(\pi_{\bar{w}}(x), \pi_{\bar{y}}(y)) \wedge (\pi_{\bar{y}}(x) \prec \pi_{\bar{w}}(y)) \wedge \neg ov(\pi_{\bar{y}}(x), \pi_{\bar{y}}(y))$$

where $PO'(x, y) := ov(x, y) \wedge (x \not\prec y) \wedge (y \not\prec x)$. $R_2(x, y)$ states that the white of the two partially overlap whereas the yolks are disjoint from each other's eggs, and $R_{11}(x, y)$ that x's white partially overlap both y's white and yolk, and x's yolk is contained in y's white.

It is also possible to combine any of the above relation-sets whenever the underlying space-lattice has a natural interpretations for each relation-set's roles. For instance, for spatio-temporal objects one could combine Allen's Interval Algebra and RCC8.

There is, of course, qualitative information that cannot be represented by our bintrees, e.g. unknown data via disjunctions, such as $EC(a, b) \vee PO(a, b)$ but where we do not know which, since our representation is a concrete model (note that we can model certain types of unknown data by introducing appropriate roles, such as done in Example 27); unions such as $a \prec b \oplus c \wedge a \not\prec b \wedge a \not\prec c$, we can only state that $ov(a, b) \wedge ov(a, c)$; space-lattices that require infinite sets of roles, such as fuzzy sets with membership-roles in $[0, 1]$; formulae with role-variables, such as $R(z, x, y) := \pi_z(x) \prec y$; or shape-relations, we have not found a way to express formulae that can state e.g. concavity.

7 Related Work

There has been done much work on efficient representations of transitive relations and structures for reachability queries in directed graphs (see e.g. [19, 27, 9]) which can be used to represent our containment relationships. However, these representations do not facilitate efficient construction or update of these structures from a set of spatial objects. They are also less expressive, as they do not have any concept similar to our roles or the $<$-ordering. In [21] the authors developed a qualitative representation of spatial data based on arrays of representative points. However, this representation has the same drawbacks as above.

There has also been done a lot of work on representing qualitative spatial information as a set of assertions in some spatial logic, whereby the main information extraction method is logical reasoning based on either logical calculi or constraint solving (see e.g. [7, 4] for an overview). These representations are more focused on complex reasoning problems rather than efficient query answering. These reasoning problems are normally at least NP-hard in general, but tractable restrictions exists (see e.g. [23, 26, 17] for RCC8) that can scale to large datasets. However, as the related work above, these approaches presupposes the existence of a constraint network, and does not themselves provide any efficient construction algorithm of these constraint networks, nor any efficient update of already constructed networks.

In [15] the authors construct a compact representation for the RCC8 and CDC (Cardinal Direction Calculus) relations over polygons using a combination of minimum bounding rectangles (MBR) for each polygon and normal relational database tables when a relation cannot be computed from the MBRs. The authors of [16] provide an efficient construction of a representations of RCC8-relationships between spatial objects via sets of rectangular pseudo-solutions. Each pseudo-solution consists of a partial interpretation of spatial objects into rectangles that encodes one part of an RCC8-network. Both of the approaches above give an efficient method for constructing their respective representations from a set of spatial objects, the former using MBRs and the latter using quadtrees. However, they are both limited to RCC8 and CDC relations over two dimensional objects, whereas our approach can handle a more expressive set of relations over elements from any space lattice.

8 Conclusion and Future Work

We have seen that we always can construct a bintree representation for any space-lattice that is correct w.r.t. any predefined set of qualitative relations expressible in our formula language. This formula language is expressive enough to express most of the common qualitative spatial relations. Our bintree representations are compact, can be stored naturally in any tuple-based representation (relational databases, triple-stores, etc.) and allow highly efficient query answering as they can be stored in a relational database and indexed by B-trees.

In the future we want to extend our implementation [11] (that currently handles all relations definable from the formulae of Definition 5) to also handle the role and order extensions and test these against real-world datasets with expressive relation-sets. We also want to compare our approach to the related representations for RCC8 and CDC described in Section 7.

It would also be interesting to try to extend the language of our relations, to for instance allow intersections, unions, or some restricted form of universal quantification in our formulae without effecting the computational properties of the representation.

References

1 S. Abiteboul, R. Hull, and V. Vianu. *Foundations of databases*, volume 8. Addison-Wesley Reading, 1995.

2 J. F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, 1983. doi:10.1016/b978-1-4832-1447-4.50033-x.

3 M. Attene, M. Campen, and L. Kobbelt. Polygon mesh repairing: An application perspective. *ACM Computing Surveys (CSUR)*, 45(2):15, 2013. doi:10.1145/2431211.2431214.

4 J. Chen, A. G. Cohn, D. Liu, S. Wang, J. Ouyang, and Q. Yu. A survey of qualitative spatial representations. *The Knowledge Engineering Review*, 30(01):106–136, 2015. doi:10.1017/S0269888913000350.

5 A. G. Cohn, J. Bennett, B.and Gooday, and N. M. Gotts. Qualitative spatial representation and reasoning with the region connection calculus. *GeoInformatica*, 1(3):275–316, 1997. doi:10.1023/A:1009712514511.

6 A. G. Cohn and N. M. Gotts. The 'egg-yolk' representation of regions with indeterminate boundaries. *Geographic objects with indeterminate boundaries*, 2:171–187, 1996.

7 A. G. Cohn and J. Renz. Chapter 13 qualitative spatial representation and reasoning. In *Handbook of Knowledge Representation*, volume 3 of *Foundations of Artificial Intelligence*, pages 551 – 596. Elsevier, 2008. doi:https://doi.org/10.1016/S1574-6526(07)03013-1.

8 A. Guttman. *R-trees: a dynamic index structure for spatial searching*, volume 14. ACM, 1984. doi:10.1145/971697.602266.

9 R. Jin and G. Wang. Simple, fast, and scalable reachability oracle. *Proceedings of the VLDB Endowment*, 6(14):1978–1989, 2013. doi:10.14778/2556549.2556578.

10 L. H. Karlsen and M. Giese. A Framework for Constructing Correct Qualitative Representations of Geometries using Mereology over Bintrees. *Annals of Computer Science and Information Systems*, 7:21–33, 2015. doi:10.15439/2015F183.

11 L. H. Karlsen and M. Giese. An Efficient Representation of Qualitative Spatial Information using Bintrees. Technical report, Department of Informatics, University of Oslo, 2017. http://hdl.handle.net/10852/53792.

12 M. Koubarakis. *Spatio-temporal databases: The CHOROCHRONOS approach*, volume 2520. Springer Science & Business Media, 2003. doi:10.1007/b83622.

13 K. Kyzirakos, M. Karpathiotakis, and M. Koubarakis. Strabon: a semantic geospatial DBMS. In *ISWC'12*, pages 295–311. Springer, 2012. doi:10.1007/978-3-642-35176-1_19.

14 G. É. Ligozat. Reasoning about cardinal directions. *Journal of Visual Languages & Computing*, 9(1):23–44, 1998. doi:10.1006/jvlc.1997.9999.

15 Z. Long, M. Duckham, S. Li, and S. Schockaert. Indexing large geographic datasets with compact qualitative representation. *International Journal of Geographical Information Science*, 30(6):1072–1094, 2016. doi:10.1080/13658816.2015.1104535.

16 Z. Long, S. Schockaert, and S. Li. Encoding large RCC8 scenarios using rectangular pseudo-solutions. In *Proceedings of the Fifteenth International Conference on Principles of Knowledge Representation and Reasoning*, pages 463–472. AAAI Press, 2016.

17 Z. Long, M. Sioutis, and S. Li. Efficient path consistency algorithm for large qualitative constraint networks. *IJCAI'16 Proceedings*, 2016. doi:10.1142/S0218213015500311.

18 Y. Manolopoulos, A. Nanopoulos, A. N. Papadopoulos, and Y. Theodoridis. *R-trees: Theory and Applications.* Springer Science & Business Media, 2010. doi:10.1007/978-1-84628-293-5.

19 P. Nicholson. *Space-Efficient Data Structures in the Word-RAM and Bitprobe Models.* University of Waterloo, 2013.

20 R. O. Obe and L. S. Hsu. *PostGIS in Action.* Manning Publications Co., Greenwich, CT, USA, 2nd edition, 2015.

21 D. Papadias and T. Sellis. The semantics of relations in 2D space using representative points: Spatial indexes. In *COSIT'93*, pages 234–247. Springer, 1993. doi:10.1007/3-540-57207-4_16.

22 D. A. Randell, Z. Cui, and A. G. Cohn. A spatial logic based on regions and connection. *KR*, 92:165–176, 1992.

23 J. Renz. Maximal tractable fragments of the region connection calculus: a complete analysis. In *IJCAI'99 Proceedings*, pages 448–454. Morgan Kaufmann Publishers Inc., 1999.

24 H. Samet. Object-based and image-based image representations. *Foundations of Multidimensional and Metric Data Structures*, pages 211–220, 2006. doi:10.1145/1031120.1031123.

25 H. Samet and M. Tamminen. Bintrees, CSG Trees, and Time. *SIGGRAPH Comput. Graph.*, 19(3):121–130, July 1985. doi:10.1145/325165.325211.

26 M. Sioutis, J. Condotta, and M. Koubarakis. An efficient approach for tackling large real world qualitative spatial networks. *International Journal on Artificial Intelligence Tools*, 25(02):1550031, 2016. `doi:10.1142/S0218213015500311`.

27 S. J. van Schaik and O. de Moor. A memory efficient reachability data structure through bit vector compression. In *SIGMOD'11 Proceedings*, pages 913–924. ACM, 2011. `doi:10.1145/1989323.1989419`.