Semantic Compression with Region Calculi in Nested Hierarchical Grids (Technical Report)

Joseph Zalewski
Kansas State University

Pascal Hitzler
Kansas State University

Krzysztof Janowicz
University of California Santa Barbara

August 28, 2021

Abstract

We propose the combining of region connection calculi with nested hierarchical grids for representing spatial region data in the context of knowledge graphs, thereby avoiding reliance on vector representations. We present a resulting region calculus, and provide qualitative and formal evidence that this representation can be favorable with large data volumes in the context of knowledge graphs; in particular we study means of efficiently choosing which triples to store to minimize space requirements when data is represented this way, and we provide an algorithm for finding the smallest possible set of triples for this purpose including an asymptotic measure of the size of this set for a special case. We prove that a known constraint calculus is adequate for the reconstruction of all triples describing a region from such a pruned representation, but problematic for reasoning with hierarchical grids in general.

1 Introduction

In traditional geographic information systems (GIS), geographic data, such as the locations of region boundaries, are stored using one of a variety of techniques based on coordinate geometry. Vertices of polygons, etc. are points in a continuous space, represented by their real-number coordinates in some coordinate system. An alternative to this is the use of so-called hierarchical grids, where "space" is subdivided into hierarchically-nested "cells", whose exact geometries can be easily computed due to their regular nature. Hierarchical grids are already long in use in GIS, being used as index structures to speed up lookup of points and objects, stored as coordinates [13]. Recently hierarchical grid systems have been employed very successfully by companies such as Google [3] and Uber [4] to structure large quantities of their internal data. While in these applications there is a strong emphasis on indexing and efficient look-up using the

index, we see another advantage to hierarchical grids that has not yet been systematically explored, namely that they lend themselves naturally to a context in which *knowledge graphs* are used for data integration and management.

Knowledge graphs are an approach to structuring data (or metadata[1]) in form of a labeled and typed graph, together with a type logic that is often referred to as a knowledge graph *schema* or an *ontology* [6]. Knowledge graphs have recently seen significant uptake by industry, with visible success [10]. The World Wide Web Consortium (W3C) has developed standards for knowledge graphs and their schemas – the Web Ontology Language OWL and the Resource Description Framework RDF – as well as the SPARQL querying language and other relevant standards, that are widely used [7]. The schema, if expressed in OWL, consists of a set of logical formulas that can be used for deductive inference if desired.

We argue that hierarchical grids are a natural choice for representing information about spatial regions, for many contexts: Each grid cell thus becomes a node in the knowledge graph, with relations between the cells (or relations between cells and features or information of interest) represented naturally by labelled graph edges. Collections of cells can be used to approximate regions of interest (e.g., by representing them with a suitable cover), thus trading some representational precision for increased querying and data processing speed. In a data integration context – in which knowledge graphs are prominently used – a chosen hierarchical grid can serve as the central integration anchor for spatial data originating from different formats, thus providing a uniform representation that can be tapped into, e.g. by visualization tools and geographical information systems.

Furthermore, type logics that provide schema information for knowledge graphs can naturally be used to capture logic-based calculi about spatial relations between regions, such as variants of the Region Connection Calculus RCC [11]. The formal logic of the region calculus and the formal logic of the knowledge graph schema then naturally combine and can be utilized for joint logical inferencing, i.e., for deducing knowledge that arises as necessary logical consequences from the data and type logic of the knowledge graph, and can for example be used for querying for logically implied, but not explicitly encoded, information.

Another interesting aspect of the combined region and type logic is that it can be utilized for what has been called *semantic compression* in the context of (RDF) knowledge graphs [8]. It refers to the idea of using logical deduction rules to compress a knowledge graph without loss of information. In some situations, for example, addition of a single suitable logical formula to the type logic can make a very large number of node-edge-node graph triples redundant in the sense that they can now be removed, while at the same time the new logical formula makes it possible to re-generate the removed triples as needed. We will return to this point later in the paper.

[1] In a knowledge graph context, the boundary between metadata and data is – deliberately – not crisp. But what is referred to as "data" in a knowledge graph context would often be called "metadata" in different contexts.

The rest of this paper is organised as follows. In Section 2 we will provide a mathematical and very general formalization of a region connection calculus on hierarchical grids. In Section 3 we provide formal and qualitative arguments why our approach aids with semantic compression. In Section 4 we provide results on the reasoning necessary to take advantage of this compression. In Section 5 we conclude the paper and discuss future work.

2 Region Connection Calculus on the Grid

We will approach region calculi and hierarchical grids from an abstract but mathematically precise vantage point. We assume that the reader is familiary with basic set-theoretic topology, see e.g. [9] and also with the basics of formal logic, see e.g. [15].

We will first define a specific grid, the *square grid*, before giving a more general definition. The *square grid* Sq is the pair $([0,1]\times[0,1], \mathsf{cells}_{\mathsf{Sq}})$ where $\mathsf{cells}_{\mathsf{Sq}}$ is a tree with root $[0,1]\times[0,1]$ and in which every node $[a,b]\times[c,d]$ has the four children

$$[a,(a+b)/2]\times[c,(c+d)/2], \qquad [a,(a+b)/2]\times[(c+d)/2,d],$$
$$[(a+b)/2,b]\times[c,(c+d)/2], \qquad [(a+b)/2,b]\times[(c+d)/2,d].$$

Let the depth-m square grid Sq_m be Sq with the tree $\mathsf{cells}_{\mathsf{Sq}}$ restricted to those nodes of distance $\leq m$ from root. Let the symbol ch denote the relation "a is a child of b in the tree $\mathsf{cells}_{\mathsf{Sq}_n}$." Use ch^* for "descendant" and ch^+ for "proper descendant." Call a cell d *minimal* if it has no children. In general, we will often refer in this paper to nodes of a tree as ordered objects; the ordering is the relation ch^*, which is a partial order.

This basic notion of a square grid is very similar to the Google S2 grid [3], and our results about it will carry over to the case of more "realistic" grids like S2 which are patched together from several square grids. We now provide a general definition of nested hierarchical grids of which the square grid is a special case.

Let a *nested hierarchical grid* be a pair (A, cells_A) where A is a topological space, and cells_A is a tree with root A and in which every node N is a nonempty topological space, and if it has children, it has finitely many, but at least two, and the children N_i of N are regular closed subspaces of N such that $\bigcup_i N_i = N$ and no two N_i share an open subset. Additionally, for this paper we will require A to be a Baire space, i.e., such that countable unions of degenerate subspaces are degenerate, which is a very mild condition given usual application scenarios for grids. In fact we really only need the condition that a finite union of degenerate sets is degenerate, but so many common spaces are Baire spaces that the distinction is not too important. For a tree T, we will use $|T|$ to denote the set of all nodes of T.

For the region calculus, we will focus on RCC5 (background in e.g. [14]) which is a formal logic defined as follows: an *RCC5 signature* is just a set of

Formula Type	**Satisfaction Condition**
$\mathsf{EQ}(a,b)$	$a^I = b^I$
$\mathsf{PP}(a,b)$	$a^I \subsetneq b^I$
$\mathsf{PP}^{-1}(a,b)$	$b^I \subsetneq a^I$
$\mathsf{PO}(a,b)$	$a^I \setminus b^I, b^I \setminus a^I, a^I \cap b^I$ each contain some open set
$\mathsf{DR}(a,b)$	$a^I \cap b^I$ has empty interior

Table 1: Satisfaction of RCC5 formulas

constants C, and the formulas are the expressions $P(a,b)$ for $a,b \in C$ where P is one of the five predicate symbols $\mathsf{EQ}, \mathsf{PP}, \mathsf{PP}^{-1}, \mathsf{DR}, \mathsf{PO}$. The semantics of RCC5 with respect to a "universe" topological space A are as follows: an *RCC5 interpretation* is a mapping $\cdot^I : C \to \mathcal{P}(A)$, such that for all $a \in C$, a^I is a nonempty regular closed set. (Recall that a regular closed set is a closed set S such that $\overline{\mathsf{int}(S)} = S$, where $\mathsf{int}(S)$ is the interior of S and $\overline{X}$ is the closure of the set X.) Often the universe chosen is the plane $\mathbb{R}^2$. Regular closed sets in the plane capture the concept of regions that have some nonzero thickness everywhere, no degenerate one- or zero-dimensional parts. A formula $P(a,b)$ is said to be *satisfied by* I ($I \models P(a,b)$) if the condition in Table 1 holds. A set of formulas Π is said to be *satisfied by* I (written $I \models \Pi$) if all its members are satisfied by I. Logical implication is defined as usual for sets of formulas Π, Γ by $\Pi \models \Gamma$ if for all interpretations I such that $I \models \Pi$, we have $I \models \Gamma$.

It can be shown that these predicates are jointly exhaustive and pairwise disjoint (JEPD), that is, for any $a,b \in C$ and interpretation I, $I \models P(a,b)$ for exactly one of the predicates P from Table 1. RCC5 is a popular "constraint calculus" used in GIS database systems [14]. However, in what follows we will provide a slightly stronger variant of RCC5, which takes into account a priori *all* information about the structure of a hierarchical grid, not just the part of that information which is expressible in RCC5.

Definition 1. *Define the logic "RCC5-G" as follows: an* RCC5-G signature *with respect to a nested hierarchical grid* (A, cells_A) *is a set of constants* C*, which we assume to be disjoint from* $|\mathsf{cells}_A|$*. The* formulas *are the expressions* $P(a,b)$ *for* $a,b \in C \cup |\mathsf{cells}_A|$*, and* P *one of the five predicate symbols* $\mathsf{EQ}, \mathsf{PP}, \mathsf{PP}^{-1}, \mathsf{DR}, \mathsf{PO}$*, as before. Semantics is defined relative to a topological space* B *such that* A *is a subspace of* B*, as follows:. An* RCC5-G interpretation *is a mapping* $\cdot^I : C \cup |\mathsf{cells}_A| \to \mathcal{P}(B)$*, such that for all* $a \in C$*,* a^I *is a nonempty regular closed set, but particularly* $d^I = d$ *for* $d \in \mathsf{cells}_A$*, as for RCC5. A formula* $P(a,b)$ *is* satisfied by I *if the condition given in Table 1 holds.*

It is usual in many geodatabase systems to use not RCC5 but the more powerful calculus RCC8. Both arise from the system RCC introduced in the paper [11], however there are many interpretations of the semantics of these calculi, and in the original paper a topological semantics is eschewed in favor of a more a-priori one. The topological semantics given here for RCC5 is merely the common-sense one, for regular closed regions. A discussion of topological

$\circ$	PP	PP^{-1}	DR	PO	EQ
PP	{PP}	{PP,PP^{-1},DR,PO,EQ}	{DR}	{DR,PO,PP}	{PP}
PP^{-1}	{PP,PP^{-1},EQ,PO}	{PP^{-1}}	{DR,PO,PP^{-1}}	{PO,PP^{-1}}	{PP^{-1}}
DR	{DR,PO,PP}	{DR}	{PP,PP^{-1},DR,PO,EQ}	{DR,PO,PP}	{DR}
PO	{PO,PP}	{DR,PO,PP^{-1}}	{DR,PO,PP^{-1}}	{PP,PP^{-1},DR,PO,EQ}	{PO}
EQ	{PP}	{PP^{-1}}	{DR}	{PO}	{EQ}

Table 2: RCC5 Composition Table.

semantics for RCC8 can be found in [12]. We use RCC5 here not just to simplify presentation, but because we believe that for our present purposes, RCC8 is actually unnecessary. Recall the approach to geometries which motivates this paper: geometries are to be considered only insofar as they can be captured by relationships to a fixed grid. The principal difference between the two RCC constraint formalisms is RCC8's concern with boundaries- it differentiates, for example, between the true disconnectedness relation and the "external connection" relation, in which two regions overlap, but in a degenerate set (with empty interior). RCC5 considers these situations to be indistinguishable (they can both provide the semantics for the predicate DR). While in traditional geometry representation schemes, the additional information about boundary relationships can be useful, our position is that for grid representation it is not. For, real-world data (locations of boundaries) should be thought of as sampled from a continuous distribution, and so it is vanishingly unlikely that a real-world boundary will ever exactly coincide with the finitely many artificial boundaries of our grid cells. Whenever this seems to happen in real data, it should by default be attributed to insufficient decimal precision, rather than assumed to have real meaning. This assumption is convenient for us, as RCC8 is not nearly as compatible with a hierarchical organization of space as RCC5, and obtaining results for it like those in the following sections is much harder.

2.1 The RCC5 Composition Table and Constraint Solving

The RCC5 composition table can be found in, e.g., [14], and for convenience we reproduce it in Table 2. It gives, for each two RCC5 relations P, Q, the largest set $P \circ Q$ of RCC5 relations R such that R^I intersects $P^I \circ Q^I$ in some interpretation I.

Note that the semantics used in [14] is set-based rather than topological; however this does not make a difference in terms of the composition table, for any reasonable kind of grid - as will be evident from the discussion in Section 4.1 below. Composition tables like this are commonly used with RCC5, RCC8 and similar systems, typically in the context of *constraint networks* (see [5]). An RCC5 constraint network C is a directed graph in which each edge is labeled by a set of RCC5 relations. A network is *atomic* if all edges are labeled by a singleton. A network is *path-consistent* if, whenever $R \in C(x, z)$, there are $P \in C(x, y)$ and $Q \in C(y, z)$ such that R is in the composition set for P and Q; and furthermore no edge is labeled by an empty set. (This is a *formal* notion of path consistency; there are also semantic notions.) In most applications of

binary constraint calculi and composition tables, reasoning tasks are focused on finding path-consistent, often atomic, networks C' that are consistent with a given network C, in the sense that $C'(x, y) \subseteq C(x, y)$ everywhere. A path-consistent network is usually used as a proxy for a network that is satisfiable with respect to some semantics. In the case of RCC5, the semantics is given by interpretations mapping nodes of C to regular closed sets, where I satisfies C if for every two nodes x, y, if $I \models P(x, y)$ according to the condition in 1, then $P \in C(x, y)$. For RCC5-G similarly, when nodes of C come from an RCC5-G signature, but interpretations must satisfy $d^I = d$ for grid cells d.

2.2 Inheritance

While in this paper we will concentrate on the use of RCC5-G to reduce sets of relations between the grid and a single region, we will briefly mention a more standard use of such calculi - to store information about properties of regions that are "upward-inherited" or "downward-inherited." By downward-inherited we mean a property which, if possessed by a region R, also characterizes all regions containing R. Such a property can be represented by the collection of all maximal regions having it, and checked by checking whether a region R is EQ or PP to one of these- that is, whether a satisfiable constraint network exists in which an edge from R to one such region is labeled $\{\mathsf{EQ}, \mathsf{PP}\}$. Often there will only be one maximal region needed, as for the property "completely covered by water." Common types of upward-inherited properties involve "containing a feature", such as a particular city, or containing some part of a distributed entity, such as "water". These can be represented by a $\{\mathsf{PP}^{-1}, \mathsf{EQ}\}$ (in the first case) or $\{\mathsf{PO}, \mathsf{PP}^{-1}, \mathsf{EQ}\}$ (in the second) relation to a region, i.e. the region occupied by the city, or the region covered by water.

3 Semantic Compression

We have already argued in the introduction that the use of hierarchical grids together with knowledge graphs, as described herein, provides some advantages in some circumstances. It is important to note, however, that that there are always trade-offs, and that a particular representational form (such as using a hierarchical grid) is advantageous in some use cases, and not so in others. Our approach provides *additional flexibility* in making a choice for representing spatial information in the context of knowledge graphs.

Using a hierarchical grid as described is an approximation for spatial representation that is constrained by the pre-defined grid cells. As such, it comes at the loss of some precision. However it also comes with some advantages. One of them is representational simplicity. Rather than representing each region with, say, a polygon in the graph, the spatial representation of the regions becomes normalized as a selection of cells that have some specified region-connection relationship to the region. By taking the hierarchical structure (and corresponding logical axiomatization) into account, it is in fact *not* necessary to flag all such

cells, as covering a region inherits upwards and containment within a region inherits downwards. We can thus arrive at a *semantically compressed* representation utilizing the logic detailed in Section 2 - and we will discuss a theorem for the square grid below, Theorem 2, that gives some formal weight to the argument that this compressed set is in fact somewhat small.

In a similar vein, not only region representation can be understood as semantically compressed, but relevant features of such a region can likewise be represented in compressed from by making use of the logic from Section 2, in particular upward and downward inheritance as discussed. E.g., if a cell is known to fully fall within a region with arid climate, then we know that arid climate also applies for all its sub-cells. In particular in the context of knowledge graphs, where information pertaining to many different regions may be abundant, this type of reasoning over the grid may result in a cleaner representation of content.

Another possible advantage of using hierarchical grids for knowledge graphs with spatial content is for the information integration process itself; indeed knowledge graphs excel as a tool for information integration from heterogeneous sources. Using a hierarchical grid, spatial information from a data source can be *normalized* by expressing it approximately using cells, thus providing a convenient format for the integrated representation, while at the same time providing a simplified logic for reasoning about spatial relations and inheritance of features as just discussed. Once cast into this form, it is no longer necessary to compute region intersections etc. from, say, vector representations, or to deal with the complexities of a region calculus on arbitrarily shaped regions: Instead we have arrived at a compressed representation with a much simpler logic.

In the spirit of compression of representation, in Theorem 1 below, we will prove correctness of an algorithm – also defined below – for computing minimal region representations in terms of cells.

The formal proofs of the theorems below are somewhat involved, but they are provided for the interested reader. We first need some preparations.

For the already mentioned Theorem 1, we consider how to arrive at a compressed representation, in terms of cells, of a single region, about which we know as much as our grid-based representation of geometry can tell us, in a vacuum, so to speak- our only option to record information about it is by RCC5 relations directly with cells, not with other regions. See the Further Work section for other similar problems we may want to solve. Given an RCC5-G signature $(A, \mathsf{cells}_A), C$, $R \in C_R$, and interpretation I, let the *full description* $\mathsf{desc}(R, I)$ of R, I be the set $\{\phi \mid I \models \phi, \phi = P(R, d) \text{ or } P(d, R)\}$ for d a cell. The *full-knowledge single region compression* problem is to find a smaller set of formulas B of the types $P(R, d)$ and $P(d, R)$, such that $\mathsf{desc}(R, I)$ is logically equivalent to B. Now in general the effectiveness of compression is hard to neatly quantify in a provable way, but in this case we can get a very nice fact - that there is an optimal solution, up to a constant discrepancy. This optimal solution is intuitively, not to say trivially, obvious to anyone who can visualize a square grid: cells which are fully inside or fully outside the region R ought to be con-

glomerated together as much as possible in B, since decomposing them into smaller cells adds no further information about where R is. Hierarchical grid libraries often contain functions to perform this kind of compression (see e.g. "compact" in H3). Note that without loss of generality we can consider only formulas of the type $P(d, R)$, since every formula $P(R, d)$ is equivalent to one of this form. Indeed, thinking and writing about the correctness of the optimal solution is very cumbersome if we keep using this predicate notation, with all its superscripts and arbitrary ordering of arguments. We introduce a different formalism, which sheds more light on the idea behind the correctness proof, and will be seen to easily generalize to certain other logics with more expressive power than RCC5-G.

3.1 The Tree-Labeling Formalism

For a tree T, let a *tree label logic* defined on T be a set of "labels" L and a set $\mathsf{Ad} \subseteq |T| \to L$ of "admissible" maps assigning labels to nodes. In particular, given an RCC5-G signature, let T be the tree of cell identifiers, defined by the ch relation, and the labels be $\mathsf{EQ}, \mathsf{PP}, \mathsf{PP}^{-1}, \mathsf{DR}, \mathsf{PO}$. Let an admissible labeling p be one for which there is a constant R, interpretation I such that if $d \in |T|, p(d) = P$, then $I \models P(d, R)$. The JEPD property implies that there is a unique such p induced by each R, I. We will call this a tree label logic induced by RCC5-G. Let a *waterfall* n with respect to an admissible labeling p be a node such that if an admissible labeling p' agrees with p at n, then $p = p'$ also on all children of n, and furthermore all children of n are waterfalls with respect to p. Clearly $\mathsf{PP}, \mathsf{EQ}, \mathsf{DR}$ have this property. Let $W \subset L$ denote the set of waterfall labels. Let a *fountain* with respect to p, similarly, be a node n such that if p' is an admissible labeling and $p(n) = p'(n)$, then p, p' agree on the siblings and parent of n, if any, and the siblings of n are waterfalls and the parent of n a fountain with respect to p. Let F denote the set of fountain labels. Say that a subset of the labels D is a set of *local* labels if, for any admissible labelings p, p', node n having at least one child, if $p(n) = l \in D$ and p', p agree on the children of n, and $p'(n) \in D$, then $p'(n) = l$ (Uniqueness); and also for any node n, if some admissible p_i assigns each child n_i of n a label in D, then there is an admissible p agreeing with p_i on all the descendants of n_i which assigns n a label in D (Existence); and also for any node n, if p, p' are admissible labelings that assign n the same label in D, there is an admissible p'' agreeing with p on nodes $\mathsf{ch}^* n$ and with p' on nodes not $\mathsf{ch}^* n$ (Locality). In the case of RCC5, the set $\{\mathsf{PP}, \mathsf{DR}, \mathsf{PO}\}$ has this property. In passing we point out that the first two parts of this definition will seem more natural if you consider the labelings whose values are drawn from a set of local labels as a sheaf. Now we will prove the correctness of a minimal generating set algorithm in this context, and apply it to the RCC5 case. Let a *generating set* for labeling p in T be a set $G \subseteq |T|$ such that, if p' is another admissible labeling which is equal to p everywhere on G, then p, p' agree everywhere on T. More generally, for a set of labels D and subset $S \subseteq |T|$, a D-generating set G on S for a labeling p, whose values on S are all in D, is a set of nodes such that if p' is another admissible labeling whose

values on S are in D, which agrees with p everywhere on G, then $p = p'$ on all nodes of T. Denote the subtree of all descendants of a node n by $\mathsf{tr}(n)$. Finding a small generating set for the labeling associated with a region constant R and interpretation I is clearly tantamount to finding a well-compressed subset of formulas logically equivalent to $\mathsf{desc}(R, I)$.

Proposition 1. *If G is a generating set for the labeling p induced by R, I, the formulas $P(d, R)$ where $p(d) = P$ are logically equivalent to $\mathsf{desc}(R, I)$.*

Proof. By definition, all these formulas are in $\mathsf{desc}(R, I)$. It remains to show that they imply $\mathsf{desc}(R, I)$. Let J be an interpretation and $P(d^J, R^J)$ hold for all $d \in G$, $P = p(d)$. Then there is a labeling p' induced by J, which agrees with p on G. Since G is a generating set, $p = p'$ on all cells d. Then since p' is induced by J, and p by I, it follows $P(d^I, R^I)$ iff $P(d^J, R^J)$. By the earlier remarks about order of arguments, $P(R^I, d^I)$ iff $P(R^J, d^J)$. So $J \models \mathsf{desc}(R^I)$, and since J was arbitrary, the set of formulas derived from G logically implies $\mathsf{desc}(R, I)$. □

Of course, this set of formulas is the same size as the set G.

Proposition 2. *The set of labels $D = \{PP, DR, PO\}$ is a set of local labels, if the RCC5-G semantics is given by $(A, \mathsf{cells}_A), B$ such that B contains a nonempty regular closed set disjoint from A.*

Proof. (Uniqueness) Let p, p' be admissible labelings of cells_A. Let $p(n), p'(n) \in D$, and $p = p'$ on the children of n. There are 3 cases: (I): $p(n) = \mathsf{PP}$. So $p(n_i) = p'(n_i) = \mathsf{PP}$ for all children n_i of n. Recall there is an interpretation I such that $p'(n_i) = P$ iff $P(n_i^I, R^I)$ holds. So all the n_i^I are subsets of R^I, so so is $n^I = \bigcup n_i^I$. Thus $p'(n) = \mathsf{PP}$ or EQ, but by hypothesis $p'(n) \in D$, so it must be PP. (II) $p(n) = \mathsf{DR}$. Same as case (I) substantially. All $p(n_i) = p'(n_i) = \mathsf{DR}$, and all n_i^I intersect R^I only in degenerate sets, and so n^I also does, by the Baire property. and $p'(n) = \mathsf{DR}$. (III): If $p(n) = \mathsf{PO}$, then at least one child n_i of n has $p(n_i) = p'(n_i)$ not PP, and at least one is not DR. So some n_i^I contains an open set in R^I and some n_i^I contains an open set not in R^I. Thus $p'(n) = \mathsf{PO}$ or PP^{-1}, but only $\mathsf{PO} \in D$.

(Existence) Let n be a node, with children n_i, and for each i an admissible labeling p_i assigning a label in D to n_i. There are 3 cases: (I): All $p_i(n_i) = \mathsf{PP}$. So all descendants of n_i also have $p_i = \mathsf{PP}$. Then let p be the labeling induced by the interpretation $R^I = B$, where B is the universe of interpretation for RCC5-G (see above). So $p(n) = \mathsf{PP} \in D$. (II): All $p_i(n_i) = \mathsf{DR}$. So all descendants of n_i also have $p_i = \mathsf{DR}$. Then let p be the labeling induced by the interpretation $R^I = C$ for C some nonempty regular closed set disjoint from A. So $p(n) = \mathsf{DR} \in D$. (III): Some child n_i has $p_i(n_i) \neq \mathsf{PP}$ and some child has $p_i(n_i) \neq \mathsf{DR}$. Let I_i be the interpretation inducing p_i. Define I by $R^I = \bigcup_i [R^{I_i} \cap n_i^{I_i}] \cup C$, for C as in case (II), and define p as the labeling induced by I. Now p must agree with p_i on descendants of n_i, and $p(n) = \mathsf{PO}$.

(Locality) We note three facts: the RCC5 relation between two regular closed sets is completely determined by the set of their *positive Venn regions* which

contain an open set, a positive Venn region of sets S_i being a set of the form $\bigcap_i V_i$, where each V_i is either S_i or $\neg S_i$, and not all V_i are $\neg S_i$. Second, if an open set intersects a regular closed set S (or its complement), it must intersect an open subset of S (resp. its complement). Third, for any set S, $\overline{\mathsf{int}(S)}$ is regular closed, and contains exactly the open sets contained in S. Now let p, p' agree at n, and R, I, I' be a region constant and interpretations inducing p, p'. The argument that p" exists as per the locality property is now straightforward but long. Define a new interpretation I'' with $R^{I''} = cl(\mathsf{int}((n \cap R^I) \cup (\neg n \cap R^{I'})))$, and let p'' be induced by this interpretation, which indeed sends R to a nonempty regular closed set. We need to prove for all cells n' that $p''(n')$ agrees with $p(n')$ or $p'(n')$, as the case may be. Case (I): $n' \,\mathsf{ch}^*\, n$. So $n' \subseteq n$. Now the open sets contained in $n' \setminus R^I$ and $n' \cap R^I$ are exactly those contained in $n' \setminus R^{I''}$ and $n' \cap R^{I''}$ respectively. Further, if there is an open set $V \subseteq R^I \setminus n'$, either V intersects n, thus intersects it in an open set, so $R^{I''}$ contains this open set; or V intersects $\neg n$, in which case there must be an open set in $R^{I'} \setminus n$ as well, because $R^I, R^{I'}$ by hypothesis must have the same RCC5 relation to n. So then $R^{I''} \setminus n'$ contains an open set. Whereas if $R^I \setminus n'$ does not contain an open set, neither can the smaller $R^I \setminus n$, so $R^{I'} \setminus n$ contains no open set, and so $R^{I''} \setminus n'$ contains no open set. So altogether we've argued that the positive Venn regions formed by $n', R^{I''}$ contain open sets iff the corresponding ones formed by n', R^I do; so the RCC5 relation between $n', R^{I''}$ is the same as for n', R^I, thus $p''(n') = p(n')$. The remaining two cases (n' disconnected from n, $n \,\mathsf{ch}^*\, n'$) recycle the same arguments. □

It is worth commenting on the awkward condition in the statement of this proposition. It amounts to saying that we cannot know what level of the hierarchy we are actually looking at, topologically, whether it is really the global level, encompassing all of space, or just a local area. This makes many arguments simpler because the case for the top of the hierarchy tree is not special.

Now let p be a D-labeling of a down-set T' of T, where D is a set of local labels. Let S (or $S(T')$) be the set of all maximal waterfalls in T' with respect to p.

Proposition 3. *S is a D-generating set for p on T'.*

Proof. Every leaf node l of T' is a waterfall node, and so there exists a maximal waterfall node above l, in S. By definition of waterfall node, any two admissible labelings agreeing on S agree on l, and thus on all leaves. Now by the Uniqueness property of D, if p, p' agree on all leaves in T', they agree everywhere, by the obvious induction on the height of a node above the leaves. □

In the context of the tree label logic induced by RCC5-G on a grid:

Lemma 1. *If $D = \{\mathsf{PO}, \mathsf{PP}, \mathsf{DR}\}$, and G is a D-generating set for p on a subtree T, for every path from the root of T to a leaf l, either $l \in G$ or some point x on the path is in G and $p(x) \neq \mathsf{PO}$.*

Proof. Note that PO has the following property: if n is a node with $m > 0$ children, $p(n) = \mathsf{PO}$, for any $m-1$ children of n, there is p' agreeing with p on n and those children, and differing on the remaining child. For trees of height 1 this is trivial. For others, a stronger claim will be proven by induction: if G is a subset of $|T|$, of height ≥ 2, and there is a path contradicting the conditions of the proposition (an "uncovered" path), there is an admissible labeling p' distinct from p on T, but equal on G and on the root. Base case: height = 2. If there is an uncovered path, the root must be labeled PO, and some child not in G. The existence of p' follows from the property of PO mentioned above. Inductive case: height > 2. First, suppose the root of T is labeled PO. (If it is not, there are no uncovered paths, and the claim is vacuously true.) Now an uncovered path in T, minus the root, is an uncovered path in some subtree $\mathsf{tr}(n_i)$ of T. So by induction we obtain p'' matching p on the portion of G in $\mathsf{tr}(n_i)$, *and* on n_i itself, but different from p. Using the Locality property on n_i we "paste together" p and p'' to obtain p' matching p on all of G and the root of T but different from p, as required. □

Note that the property of PO-nodes used in this lemma is not very special to PO-nodes; in general it is the definition one should make for an "anti-waterfall", which provides sufficiently little information on nodes below itself that it might as well provide no information.

In the context of the tree label logic induced by RCC5-G on a grid, in which each non-leaf cell has at least 2 children:

Proposition 4. *For T' be a subtree of T, p an admissible labeling which takes D-values on T', $S(T')$ is of minimal size among D-generating sets for T'.*

Proof. Let G be any D-generating set for p on T'. Let w be a maximal waterfall in T'. If w is a leaf, since no non-PO label can exist on the path from w up to the root, or it would not be maximal, the above lemma implies $w \in G$. If w is not a leaf, there are at least two paths a, b from the root to leaves through w, and both must have points x_a, x_b of G below w, by the above lemma. However, clearly $G \setminus \{x_a, x_b\} \cup \{w\}$ is still a generating set, since w is a waterfall, and this set is smaller than G. So we can assume all minimal generating sets contain all maximal waterfalls, that is, are supersets of S. It follows that S is of minimal size, and in fact is the unique generating set of minimal size. □

Proposition 5. *Let (T, L) be a tree label logic. If p is an admissible labeling of T and there is some fountain node with respect to p, then either there is a waterfall fountain node or there is a least fountain node in T.*

Proof. Let f be a fountain node. All nodes must be descendants of f, ancestors of f or waterfalls: let n be another node. We will prove the claim by induction on the shortest distance d from f, n to their least common ancestor a. If $d = 0$, either $a = f$ or $a = n$, and so either n is a descendant of f or an ancestor of f, as required. Let $d > 0$, and assume the claim is true for smaller distances. First let f be closer to a than n. Call f's parent f'. Then we can assume n is an ancestor or descendant of f' or a waterfall. If an ancestor, then it is an ancestor of f; if a

waterfall, the claim is also proven. If a descendant of f', then it is a descendant of a sibling of f, but those are waterfalls, so n is a waterfall. Second, let n be closer to a than f. Call n's parent n'. Then we can assume n' is an ancestor or descendant of f or a waterfall. If a descendant, then n is a descendant of f; if a waterfall, so is n; if an ancestor, then n' must be a fountain, and have some child of which f is a descendant, but since $d > 0$, this child isn't n. So n is a sibling of a fountain and therefore is a waterfall. It follows from this claim that if f, f' are two fountain nodes that are not comparable in the tree, both must be waterfalls. So all non-waterfall fountain nodes must be comparable, i.e. exist along a single path in the tree, and so there is a least such node. □

Let (T, L, Ad) be a tree label logic in which no node of T has more than k children, and in which L is partitioned into a set of fountain labels and a set of local labels D. The following proposition and algorithm concern this situation, of which RCC5-G is a special case.

Proposition 6. *Let p be an admissible labeling of T, and f a minimal fountain node of p. A minimal generating set for p cannot be smaller than a minimal D-generating set for the nodes strictly below f, except by a constant discrepancy $\leq k$.*

Proof. Clearly all nodes strictly under f are labeled by p with D-labels, since otherwise they would be fountains, contradicting the minimality of f. Without loss of generality, if G is a minimal generating set, we can assume that the only node in G that is not strictly below f is f itself: if there are some others, they can be replaced with f, not increasing the size of G, and the generating set property is not lost, since $\{f\}$ generates the values of p on all nodes not strictly below f. Now let G' be the portion of G that is strictly below f. $G' \cup \{n \mid n \,\mathsf{ch}\, f\}$ is a D-generating set for p on the set of nodes strictly below f: for contradiction, let p' be an admissible labeling that takes D-values below f, equal to p on $G' \cup \{n \mid n \,\mathsf{ch}\, f\}$, but unequal somewhere below f. Using the locality property repeatedly on the children of f, get an admissible labeling p'' which agrees with p everywhere not below f, and in particular on f itself, but with p' strictly below f. But now p'' agrees with p on G, and not everywhere, which contradicts that G is a generating set. Now let H be a minimal D-generating set for p on the nodes below f. $|H| \leq |G' \cup \{n \mid n \,\mathsf{ch}\, f\}| \leq |G'| + k \leq |G| + k$, so $|G| \geq |H| - k$, as required. □

Theorem 1. *The algorithm in Figure 1 correctly computes a minimal generating set up to a discrepancy bounded by the maximum branching factor of T.*

Proof. First, in case the "if" in line 1 is taken, $\{f\}$ is a generating set, and it is certainly no more than 1 larger than the smallest such set. Now, if the "if" is not taken, in line 3 S is assigned a set of proper descendants of b; whenever S has this property, so does $\mathsf{refine}(S)$. We will prove that $\mathsf{refine}^\omega(S)$ is the minimal D-generating set for the nodes below f.

The following abstract algorithm computes the set S. It is an abstract algorithm because it relies on being able to check whether a node is a waterfall or is a fountain. This is not hard to do for the case of RCC5-G and similar systems.

Input: Tree T, admissible labeling p
1 if there is a waterfall fountain node f in T **return** $\{f\}$
2 f := the least fountain node in T
3 S := $\{n \mid n \,\mathsf{ch}\, f\}$
4 while S != refine(S): S := refine(S)
5 return S $\cup\{b\}$
where
refine(S) = $[\bigcup_{s\in S\setminus W} \{c \mid c \,\mathsf{ch}\, s\}] \cup [W \cap S]$

Figure 1: An Abstract Algorithm to Compute a Small Generating Set

Next, that $\mathsf{refine}^\omega(S)$ is well-defined: let the height of a node x be defined as the maximum distance in T from x to a descendant of x that is in W. Let the height of a set of nodes S be the maximum height among its members. We claim that if S has height n, $\mathsf{refine}^{n+1}(S) = \mathsf{refine}^n(S)$. By induction on n: if $n = 0$, then all nodes in S are in W. It is then plain to see that $\mathsf{refine}(S) = S = \mathsf{refine}^0(S)$. Inductive step: Suppose the height of S is $n > 0$. Let $x \in \mathsf{refine}(S)$. Either $x \in W$, so its height is 0, or $x \,\mathsf{ch}\, y \in S$ with positive height $m \leq n$, but then x must have height no more than $m - 1$, so the height of $\mathsf{refine}(S)$ is bounded by $n - 1$, thus $\mathsf{refine}^n(\mathsf{refine}(S)) = \mathsf{refine}^{n-1}(\mathsf{refine}(S))$, as needed. Thus the loop in line 4 terminates, and in steps bounded by the depth of T.

Let T' denote the down-set $\{n \in |T| \mid n \,\mathsf{ch}^+ f\}$. Now we need to prove that $\mathsf{refine}^\omega(S)$ is indeed the set $S(T')$. Let w be a maximal waterfall in T'. Since S contains all children of f, it surely contains an ancestor of w, at a distance m from w. Let S' be any set with this property. Then $\mathsf{refine}^\omega(S')$ contains $S(T')$: Let M be the greatest distance to an ancestor in S', for all maximal waterfalls w. In case $M = 0$, $S(T') \subseteq S'$. Otherwise, by definition of refine, if w has a distance of m to its nearest ancestor a in S', this ancestor cannot be $\in W$, so $\mathsf{refine}(S')$ contains a child of a that is an ancestor of w, thus at a distance only $m - 1$. It follows that the maximal distance for $\mathsf{refine}(S')$ is $M - 1$. Further, $\mathsf{refine}(S')$ still has the property that it contains an ancestor of w; for a child of a proper ancestor is an ancestor, and if no proper ancestor of w is in S', w is, and thus $w \in \mathsf{refine}(S')$. By induction, $S(T') \subseteq \mathsf{refine}^\omega(\mathsf{refine}(S')) = \mathsf{refine}^\omega(S')$.
It remains to show that $\mathsf{refine}^\omega(S) \subseteq S(T')$. Note that the initial set S has the property that every node is above a maximal waterfall (recall a leaf is automatically a waterfall). We show that for any such set S', $\mathsf{refine}^\omega(S) \subseteq S(T')$. Let $M(S')$ be the greatest distance from a node of S' to a maximal waterfall under it. If $M = 0$, clearly $S' \subseteq S(T')$. For the inductive case: $\mathsf{refine}(S')$ still has the property of S': for if $x \in \mathsf{refine}(S')$, and is a waterfall, either $x \in S'$, so is maximal, or x's parent is in S', and is not a waterfall nor is any ancestor of it in T', so x is maximal. Whereas if x is not a waterfall, its parent is in S' and

so no waterfall exists in T' above x (but there must be one below, trivially), so there is a maximal waterfall below x. Further, $M(\mathsf{refine}(S')) = M(S') - 1$. For if there is a non-zero-length path from x to a maximal waterfall w below x, x is not a waterfall, and so its parent is in S' and has a path one longer to w. Now by inductive hypothesis $\mathsf{refine}^\omega(S') = \mathsf{refine}^\omega(\mathsf{refine}(S')) \subseteq S(T')$. □

Proposition 7. *It is possible to check whether a node is a waterfall or fountain in an RCC5-G-labeling.*

Proof. A node n is a waterfall iff it is a leaf or $p(n) \in \{\mathsf{EQ}, \mathsf{PP}, \mathsf{DR}\}$. For a nonleaf labeled PO or PP^{-1}, clearly there are at least two distinct labelings of its descendants. Likewise, n is a fountain iff it is the root or $p(n) \in \{\mathsf{EQ}, \mathsf{PP}^{-1}\}$. Otherwise, clearly there are at least two admissible labelings distinct on its parent. □

3.2 Size of Compressed Sets

It is of interest to us to know, even though this compression is optimal, how much the size of the region description is actually reduced by using it. There is a clean answer to this question for rectangular regions that fit neatly into the square grid. Let a *regular rectangle* in the grid Sq_n be a region $R = [a, b] \times [c, d] \subseteq [0, 1] \times [0, 1]$ such that R can be covered exactly by cells of Sq_n. Let the *perimeter* n of R be defined as the number of minimal cells that contact the boundary of R. A generic regular rectangle of perimeter n obviously contains $\Theta(n^2)$ minimal cells, and so $\Theta(n^2)$ total cells, since there are more minimal subcells in any given cell than there are other subcells (because $4^n > \sum_0^{n-1} 4^i$). (A function is said to be in $\Theta(n)$ if it is in $O(n)$ and $\Omega(n)$; see a standard reference such as [1]).

Theorem 2. *A regular rectangle of perimeter n can be covered exactly with $\Theta(n)$ cells in the square grid.*

Proof. First, we will prove a related fact. Define the *one-dimensional square grid* Lq as the interval $[0, 1]$ with the tree $\mathsf{cells}_{\mathsf{Lq}}$ containing root $[0, 1]$ and for each node $[a, b]$, children $[a, (a + b)/2], [(a + b)/2, b]$. Define Lq_n analogously to Sq_n. Let a *regular interval* in Lq_n be $[a, b] \subseteq [0, 1]$ which can be covered exactly by cells of Lq_n. Let the *length* of a regular interval be the number of minimal cells contained in it. In this proof we will abuse the notation $I \setminus J$ for $\overline{I \setminus J}$, since matters of topology are irrelevant to the argument. Claim: a regular interval I of length n can be divided into two intervals each exactly covered by a set of cells in which no more than one cell of a given depth in $\mathsf{cells}_{\mathsf{Lq}}$ occurs ("binary covering"). First note that a cell of depth $n - m$ in $\mathsf{cells}_{\mathsf{Lq}_n}$, henceforth "$m$-cell", is a regular interval of length 2^m. Choose a maximal m such that I contains an m-cell, and distinguish 2 cases - I contains exactly one such m-cell, or two m-cells. Three or more is impossible - it is easily seen that an interval containing three m-cells contains an $m + 1$-cell, contradicting the maximality of m. Case 1 (one m-cell c): let I_1, I_2 be the two intervals (possibly empty) comprising $I \setminus c$. Since they are adjacent to an m-cell, each contains an m-cell

iff it has length $\geq 2^m$, therefore both $|I_i| < 2^m$. Case 2 (two m-cells c_1, c_2): c_1, c_2 are adjacent, for otherwise a third m-cell would exist between them. So $I \setminus c_1 \setminus c_2$ is again a union of two intervals I_i, and by the same arguments above they each have length $< 2^m$ and share an endpoint with an m-cell, that is, a cell at least as long as I_i itself ("endpoint property"). It now suffices to prove that these intervals have binary coverings, and a binary covering of two divisions of I is obtainable by using c (resp. c_1, c_2) together with the coverings of I_i to cover the subintervals $I_1 \cup c$, I_2 (resp. $I_1 \cup c_1, I_2 \cup c_2$; assume I_i, c_i appear "left to right"). These will be binary coverings because I_i do not contain any cells of the same order as $c_{(i)}$ for reasons of sheer size. Now, let I have the endpoint property. Assume as inductive hypothesis that all interval of length $\leq 2^n$ with the endpoint property have binary coverings. The base case $n = 0$ clearly holds. Observe that I contains an m-cell iff I contains an m-cell adjacent to the large cell witnessing the endpoint property iff $|I| \geq 2^m$. So choose maximal m such that I contains an m-cell c, and $|I \setminus c| < |I|/2$, else m would not be maximal. By hypothesis, a binary covering exists for $I \setminus c$, and it contains no m-cell, so this covering together with c is a binary covering of I.

Returning to two dimensions, note that any regular rectangle $R = [a, b] \times [c, d]$ of perimeter p in Sq_n induces two regular intervals $[a, b], [c, d]$ in Lq_n, with length $O(p)$, p the perimeter of R. Now dividing these into binary-covered intervals as above yields a partition of R into four smaller rectangles, each of perimeter $\leq p$. Choose one of these, $R' = [x, y] \times [z, w]$, and let C, D be binary coverings of $[x, y], [z, w]$. Define a covering of R' by smaller rectangles $\{[a', b'] \times [c', d'] \mid [a', b'] \in C, [c', d'] \in D\}$. Each such rectangle is the product of an m-cell and an $m + i$-cell, so can itself be covered by 2^i two-dimensional cells. Dividing each rectangle in this way, we obtain a covering of R' by cells. How many are there? We need to count the number of $1 : 2^n$ rectangles for each n. Let k be the largest order of cell appearing in C or D. $k \leq \log(O(p))$. There cannot be more than $2 * (k - n + 1)$ rectangles of ratio $1 : 2^n$, since each requires either an m-cell in C and an $m + n$-cell in D or vice versa, and only 1 of each can exist, by hypothesis. So the total number of cells in our covering of R is bounded by a multiple of $\sum_0^k 2^n(k - n + 1)$. We obtain

$$\begin{aligned}
\Delta \sum_0^k 2^n(k - n + 1) &= \sum_0^{k+1} 2^n(k - n + 2) - \sum_0^k 2^n(k - n + 1) \\
&= 2^{k+1}(k - (k + 1) + 2) + \sum_0^k 2^n(k - n + 2) - \sum_0^k 2^n(k - n + 1) \\
&= 2^{k+1} + \sum_0^k 2^n(1) = 2^{k+2} - 1 \leq 2^{k+2},
\end{aligned}$$

whereas $\Delta(8 * 2^k) = 2^{k+2}$, so it follows that our quantity is in $O(2^k)$, and thus $O(2^{(\log O(p))}) = O(p)$. Since each of the 4 parts into which we split R can be covered by $O(p)$ cells, so can R.

For the lower bound, let R in Sq_n be $[0, 1] \times [0, 1/2^n]$. This set cannot contain

a cell larger than a 0-cell, but is 2^n times longer than a 0-cell; clearly it requires 2^n cells to cover, which is proportional to its perimeter. □

It should be noted that this theorem does not imply that there is for every regular rectangle R^I of perimeter p a set of $\Theta(p)$ RCC5-G formulas equivalent to $\mathsf{desc}(R, I)$, because depending on its position, many formulas (with predicate DR) may be needed to describe the place where R is not. However, as long as R is contained in a cell not too much larger than itself, such a set will exist.

4 Semantic Decompression

In this section, we discuss how reasoning over the RCC5-G calculus can be accomplished, first developing technical background in a more general setting and then applying the RCC5 composition table to the problem of recovering a region's full description from its compressed representation.

4.1 First-Order Reasoning with RCC5-G

In order to facilitate reasoning with RCC5-G, we present a translation to first-order logic preserving consequences, for sufficiently well-structured grids.

RCC5-G can be considered an application of a topological propositional logic. Say that two subsets in a topological space are equivalent up to degeneracy if their symmetric difference is degenerate. This is obviously a reflexive and symmetric relation. In a Baire space, it is also transitive. Thus we will write it $\cong$. For a set V of propositional variables, and a topological space A, let a *topological assignment* t be a mapping from FV (the set of formulas in $\{\wedge, \vee\}$ over V) to $\mathcal{P}A$, such that for all formulas ϕ, ψ, $t(\neg\phi) \cong A \setminus t(\phi)$, and $t(\phi \wedge \psi) \cong t(\phi) \cap t(\psi)$. Say that a topological assignment t satisfies a formula ϕ if $t(\phi)$ is nondegenerate. Let a *Venn formula* η over V be a formula of the form $\bigwedge_v s_v$ where for each $v \in V$, s_v is either v or $\neg v$. (Assume $\bigwedge$ is defined so as to make this an unambiguous specification of a formula.)

Proposition 8. *If two formulas ϕ, ψ are classically equivalent, and t is a topological assignment into a Baire space, then $t(\phi) \cong t(\psi)$.*

Proof. For any formula $\phi(\vec{x})$ of n variables, perhaps not using all of them, set S, let B_ϕ be the operator : $S^{\vec{x}} \to S$ induced by ϕ in the obvious way - $B_{\phi(\vec{x})\wedge\psi(\vec{x})} = B_\phi \cap B_\psi$, etc. Two formulas ϕ, ψ are classically equivalent iff B_ϕ, B_ψ are the same function. We prove by induction that for any topological assignment t, $t(\phi(\vec{x})) \cong B_\phi(t[\vec{x}])$. Base case: $\phi(\vec{x})$ is a single variable in $\vec{x}$. So B_ϕ picks out the set indexed by v. Now $t(\phi(\vec{x})) = t(v) = B_\phi(t[\vec{x}])$. Inductive case: $\phi = \neg\psi$: $t(\neg\psi) \cong A \setminus t(\psi) \cong A \setminus B_\psi(t[\vec{x}]) = B_\phi(t[\vec{x}])$, where the first $\cong$ is by definition of topological assigment and the second by inductive hypothesis. Inductive case: $\phi = \phi_1 \wedge \phi_2$. $t((\phi_1 \wedge \phi_2)(\vec{x})) \cong t(\phi_1) \cap t(\phi_2) \cong B_{\phi_1}(t[\vec{x}]) \cap B_{\phi_2}(t[\vec{x}]) = B_\phi(t[\vec{x}])$. Now if ϕ, ψ are classically equivalent, choosing some $\vec{x}$ that includes both their variables, $t(\phi) \cong B_\phi(t[\vec{x}]) = B_\psi(t[\vec{x}]) \cong t(\phi)$, and the proposition is finished. □

Proposition 9. *Let V be a finite set of propositional variables and $t : FV \to \mathcal{P}A$ be a topological assignment, for A a Baire space. Then there is a set S and classical propositional assignment $t' : FV \to \mathcal{P}S$ such that for all $\phi \in FV$, t' classically satisfies ϕ iff t satisfies ϕ in the topological sense.*

Proof. Let $S = \bigcup_\eta \mathsf{int}(t(\eta))$, where η ranges over Venn formulas. It is easy to show that all the distinct $t(\eta)$ have degenerate intersection. Therefore their interiors cannot overlap, so all the $\mathsf{int}(t(\eta))$ are disjoint. Now define $t'(\phi) = \bigcup_\eta \mathsf{int}(t(\eta))$, η ranging over all Venn formulas such that $\eta \to \phi$ is a classical tautology. t' is a classical assignment: $\eta \to \phi$ is a tautology iff $\eta \to \neg\phi$ is not a tautology, so $t'(\phi)$ and $t'(\neg\phi)$ are true complements in S. Likewise $\eta \to \phi \wedge \psi$ is a tautology iff $\eta \to \phi$ and $\eta \to \psi$ are. So $t'(\phi \wedge \psi)$ is the intersection of $t'(\phi)$ and $t'(\psi)$. It remains to show $t'(\phi)$ is nonempty iff $t(\phi)$ is nondegenerate. By 8, $t(\phi) \cong t(\mathsf{DNF}(\phi))$, which is a disjunction of all the η that tautologically imply ϕ. So $t(\phi)$ is degenerate iff $t(\mathsf{DNF}(\phi))$ is, iff all η implying ϕ have $t(\eta)$ degenerate, because of the Baire property. In this event, $t'(\phi)$ is a union of empty interiors, and is empty. But if some η is not degenerate, its interior is nonempty, so $t'(\phi)$ is nonempty, as needed.

□

Proposition 10. *Let G, V be distinct finite sets of propositional variables, and $t : FG \to \mathcal{P}A$ a topological assignment into a Baire space A, such that $t(g)$ is regular closed for all $g \in G$. Let $t' : F(G \cup V) \to \mathcal{P}S$ for some set S be a classical assignment, such that for formulas $\phi \in FG$, $t' \models \phi$ iff $t \models \phi$. Further suppose that for every G-Venn formula η and natural number n the closure of $t(\eta)$ is equal to a union of n nondegenerate regular closed sets with degenerate overlaps. Then there is a topological assignment $q : F(G \cup V) \to \mathcal{P}A$ such that $q(g) = t(g)$ for $g \in G$, all $q(x)$ for $x \in (G \cup V)$ are regular closed, and for all formulas ϕ over $G \cup V$, $q \models \phi$ iff $t' \models \phi$.*

Proof. Note that for any G-Venn formula η, there are $2^{|V|}$ $(G \cup V)$-Venn formulas η' whose disjunction is tautologically equal to η. For each nondegenerate $t(\eta)$, let n be the number of $\eta' \to \eta$ such that $t'(\eta')$ is not empty, choose a covering of $t(\eta)$ by n regular closed sets, and assign each nonempty η' implying η a unique block $r(\eta')$ of the covering of η. When $t(\eta)$ is degenerate, let $r(\eta') = \emptyset$. Let $q(\phi) = \cup_{\eta'} r(\eta')$ ranging over all $\eta' \to \phi$. Now let $t' \models \phi$. So $t' \models \mathsf{DNF}(\phi)$, and so $t' \models \eta'$ for some $\eta' \to \phi$. Thus η''s G-restriction η is also nonempty in t', so $t \models \eta$, and since $t(\eta)$ is nondegenerate, it is partitioned into regular closed sets and $r(\eta')$ is nondegenerate. Thus $q(\phi)$ is nondegenerate. Now suppose $t' \not\models \phi$. So $t' \not\models \eta$ for all $\eta \to \phi$, so all the $t(\eta)$ are degenerate, and for all $\eta' \to \phi$, $q(\eta') = \emptyset$. Now we must show that q is a topological assignment. Note that every two $q(\eta')$ for nonequivalent η' have degenerate overlap. So by the Baire property, $\bigcup_{\eta'} r(\eta') \mid \eta' \to \phi$ and $\bigcup_{\eta'} r(\eta') \mid \eta' \to \neg\phi$ have degenerate overlap. Their union covers the union of all the $r(\eta')$, which covers all but degenerately much of A, as can be perceived using 8, for example. So the first axiom of a topological assignment is satisfied. Likewise $\bigcup_{\eta'} r(\eta') \mid \eta' \to \phi \wedge \psi = \bigcup_{\eta'} r(\eta') \mid \eta' \to \phi \cap \bigcup_{\eta'} r(\eta') \mid \eta' \to \psi$. Finally, for $g \in G$, observe that for all the

RCC5 Relation	Satisfied	Not Satisfied
$\mathsf{EQ}(a,b)$	$a \wedge b$	$a \wedge \neg b$, $b \wedge \neg a$
$\mathsf{PP}(a,b)$	$a \wedge b$, $b \wedge \neg a$	$a \wedge \neg b$
$\mathsf{PP}^{-1}(a,b)$	$a \wedge b$, $a \wedge \neg b$	$b \wedge \neg a$
$\mathsf{DR}(a,b)$	$b \wedge \neg a$, $a \wedge \neg b$	$a \wedge b$
$\mathsf{PO}(a,b)$	$a \wedge b$, $a \wedge \neg b$, $b \wedge \neg a$	

Table 3: Propositional translation of RCC5-G.

nondegenerate $t(\eta)$ with $\eta \to g$, $\overline{t(\eta)}$ is a union of finitely many regular closed sets and thus is regular closed, as is $t(g)$ by hypothesis, and $\bigcup t(\eta) \cong t(g)$, so if any point $x \in t(g)$ and x is in an open neighborhood V, V intersects the interior of $t(g)$, thus intersects some $t(\eta)$, because $V \cap \mathsf{int}(t(g))$ cannot be contained in a degenerate set. Thus x is a cluster point of $\bigcup t(\eta)$, and it follows $t(g) = \bigcup \overline{t(\eta)} = \bigcup r(\eta')$ over the $\eta' \to g$, $= q(g)$. □

Note that the covering property required by this proposition is intuitively true when G consists of square grid cells or similar and $t(g) = g$.

The above two propositions allow RCC5-G to be reduced to first-order logic. First note that the RCC5 relations can be defined by simultaneous satisfiability of certain formulas by a topological assignment (in which all variables are sent to regular closed sets.) For example, $\mathsf{PO}(a,b)$ can be expressed by the existence of an assignment satisfying $a \wedge \neg b$, $a \wedge b$, $b \wedge \neg a$ simultaneously. So a finite set F of RCC5-G relation axioms can be transformed to a finite set F' of propositional formulas, and iff F was RCC5-G satisfiable, F' is satisfiable by a topological assignment sending grid cell variables to grid cells exactly. Call the subassignment which acts on the grid cell variables the *grid structure assignment*. Now consider the same set F' in light of the classical semantics. If F' is satisfiable in the way just described, by 9, it is classically satisfiable. But if F' is classically satisfiable and consistent with the set of formulas concerning grid cells that are true in the grid structure assignment, then by proposition 10 F' is satisfiable by a topological assignment which respects the grid structure. So if we can enumerate a set of axioms for a given grid which determine all formulas about the grid cells themselves, we can solve the RCC5-G satisfiability problem by solving a classical satisfiability problem. This simultaneous satisfiability of propositional formulas can be further reduced to ordinary satisfiability in first-order logic, by replacing each propositional formula ϕ with a single-variable predicate formula $\Phi(x)$ (using uppercase letters), each $\phi \wedge \psi$ by $\Phi(x) \wedge \Psi(x)$, and each $\neg\phi$ by $\neg\Phi(x)$, recursively. Now to express satisfaction of each of several ϕ_i and non-satisfaction of each of several ψ_j simultaneously one can use satisfiability of the predicate formula $\bigwedge_i \exists x \Phi_i(x) \wedge \bigwedge_j \forall x \neg \Psi_j(x)$.

The translation of RCC5-G into propositional logic proceeds indicated in Table 3. We trust these relations will be sufficiently obvious, in light of the JEPD property. Note that a nested hierarchical grid can be completely described by formulas of the form $c = \bigvee_i c_i$ for cells c and child cells c_i. The only variation possible is in how many children each cell has.

4.2 Decompression by Constraint Solving

The use of the RCC5 composition table for reasoning *with grids* is not in general very powerful. It is not complete with respect to the RCC5-G semantics, in the following sense:

Theorem 3. *Given an RCC5-G signature* $(A, \mathsf{cells}_A), C$*, there may be an atomic RCC5(-G) constraint network with nodes the constants of* C *which is path-consistent but not satisfiable.*

Proof. For instance, consider a grid with one parent cell c and two children c_1, c_2, and a single region constant R. The following network C is path-consistent but not satisfiable in the RCC5-G sense (assume whenever $P \in C(x, y)$, $P^{-1} \in C(y, x)$, see table 4) :

$$C(c_1, c) = \{\mathsf{PP}\} \tag{1}$$
$$C(c_2, c) = \{\mathsf{PP}\} \tag{2}$$
$$C(c_1, c_2) = \{\mathsf{DR}\} \tag{3}$$
$$C(c_1, R) = \{\mathsf{PP}\} \tag{4}$$
$$C(c_2, R) = \{\mathsf{PP}\} \tag{5}$$
$$C(c, R) = \{\mathsf{PO}\} \tag{6}$$

That this network is path-consistent can be readily seen by the fact that it has an RCC5 model (which is not an RCC5-G model). In an RCC5-G interpretation, since $c_1^I, c_2^I \subseteq R^I$, so is $c^I \subseteq R^I$, not partially overlapping R^I. □

The essential problem is that binary relations alone cannot readily capture the idea that the children of c cover c, together but not individually. This "problem" cannot be easily avoided. However, the RCC5 composition table is in a certain way complete for the specific purpose we would put it to in this paper, that is, to decompress a representation of a region which has been compressed. Just as we can alternatively represent sets of RCC5 formulas describing a single region constant R as labelings of a subset of a tree, we can also represent them as a constraint network. Assume the RCC5-G signature satisfies the covering condition from Proposition 10, to ensure that the composition table is correct. Let F be a set of RCC5-G formulas containing a single region constant R. Let the nodes of the network N be the cell constants of the RCC5-G signature and R, and let there be an edge between every two nodes, in both directions. Let the edge from c to d be labeled $\{\mathsf{PP}\}$ (and the reverse $\{\mathsf{PP}^{-1}\}$) when c is a child cell of d, and $\{\mathsf{DR}\}$ when c is a sibling cell of d. Let the edge $d \to R$ be labeled $\{P\}$ whenever $P(d, R) \in F$, and likewise for $R \to d$. Also, let the reverse edge be labeled $\{P^{-1}\}$, as defined in table 4. These labelings are obviously logical consequences of F as well. Let all other edges be labeled with all the RCC5 relations.

Theorem 4. *If* F *is the set of formulas obtained from the minimal generating set of Theorem 1, there is exactly one path-consistent atomic network* N' *such that for all* x, y*,* $N'(x, y) \subseteq N(x, y)$*.*

Predicate P	Inverse P^{-1}
PP	PP^{-1}
PP^{-1}	PP
EQ	EQ
DR	DR
PO	PO

Table 4: Inverse RCC5 relations.

Computing this network can be accomplished using well-developed standard tools.

Proof. First, note that there must be some path-consistent atomic labeling, since F is a satisfiable set, being equivalent to some $\mathsf{desc}(R, I)$ for interpretation I, so I can be considered an interpretation of N as well, and induces a path-consistent N' by letting $N'(x, y)$ be the RCC5 relation which holds between x^I, y^I, according to table 1.

It remains to establish uniqueness. Let N' be a path-consistent atomic network on the same nodes as N such that $N'(x, y) \subseteq N(x, y)$ for all x, y. So whenever $N(x, y)$ is a singleton, $N'(x, y) = N(x, y)$. First for the cells: if c, d are two distinct cells, either one is a descendant of the other, or they have a common ancestor. Wlog assume $c\,\mathsf{ch}^*\,d$. If $c\,\mathsf{ch}\,d$, then by hypothesis $N'(c, d) = \{\mathsf{PP}\}$. Otherwise, by induction, if e is the parent of d, $N'(c, e) = \{\mathsf{PP}\}$, and $N'(e, d) = \{\mathsf{PP}\}$ by hypothesis, so if N' is path consistent, $N'(c, d) \subseteq \{\mathsf{PP}\}$, because $\mathsf{PP} \circ \mathsf{PP} = \{\mathsf{PP}\}$ in the RCC5 composition table, and because N' is path-consistent, $N'(c, d) \subseteq \mathsf{PP} \circ \mathsf{PP}$ and $N'(c, d)$ must not be empty. So $N'(c, d) = \{\mathsf{PP}\}$. We trust that these kind of arguments are obvious and will not say them in such detail from now on. If c, d are incomparable, let e be their least common ancestor; if $c, d\,\mathsf{ch}\,e$, $N'(c, d) = \{\mathsf{DR}\}$ by hypothesis, and now using the fact $\mathsf{PP} \circ \mathsf{DR} = \{\mathsf{DR}\}$ and induction in the length of the path from c to d, $N'(c, d) = \{\mathsf{DR}\}$.

Now for the edges involving R: let p be the tree labeling induced by I, as in section 3.1. There is a generating set G for P which contains exactly the cells c which occur in formulas of F. Now, if any cell d has $N'(d, R) = \{\mathsf{PP}\}$ or $\{\mathsf{DR}\}$ or $\{\mathsf{EQ}\}$, then there is a unique singleton $N'(d', R)$ and $N'(R, d')$ for all $d' \leq d$, by arguments like the above, and the element of $N'(R, d')$ is the inverse of that in $N'(d', R)$. Similarly for $N'(f, R) = \{\mathsf{EQ}\}$ or $\{\mathsf{PP}^{-1}\}$, there are unique inverse singletons $N'(d', R)$ and $N'(R, d')$ for all $d' \not< f$. In case G contains a node labeled EQ, this is enough to establish unique atomic N'. Otherwise, there is a least fountain node f, $N'(f, R) = \{\mathsf{PP}^{-1}\}$ or f is the root of cells_A, and uniqueness is established for all nodes $\not< f$. Now for nodes that are $< f$, recall the properties of the set G: every leaf node $l < f$ is $\leq$ some maximal waterfall node; that is, w such that $N'(w, R) = \{\mathsf{DR}\}$ or $\{\mathsf{PP}\}$, and in this case $p(l) = \mathsf{DR}$ (resp. PP), or else l itself is in G, and $N'(l, R) = \{p(l)\}$. Therefore N' is uniquely determined as $N'(l, R) = \{p(l)\}$ on all leaf nodes. It is not hard to see that also $N'(R, l) = \{p(l)^{-1}\}$. Now let d

be a node $< f$ of distance $n > 1$ from the furthest leaf $l \leq d$. If all children d' of d have $p(d') = \mathsf{PP}$ or all have $p(d') = \mathsf{DR}$, then appealing to I we see that $p(d) = \mathsf{PP}$ (resp. DR), so no maximal waterfall can be $< d$, because d itself is a waterfall in p. Thus some maximal waterfall w (with label PP or DR) is $\geq d$, so $N(w, R) = \{p(w)\}$, and by the above, $N'(d', R) = \{p(w)\} = \{p(d')\}$ for $d' \,\mathsf{ch}\, d$, and since $d \,\mathsf{ch}^*\, w$ as well, $N'(d, R) = \{p(w)\} = \{p(d')\} = \{p(d)\}$, and $N'(R, d)$ contains the inverse. The next inductive case: some child d' of d has $p(d') = \mathsf{PO}$. By induction $N'(R, d') = \{\mathsf{PO}\}$, and $\mathsf{PO} \circ \mathsf{PP} = \{\mathsf{PO}\}$, so $N'(R, d) = \{\mathsf{PO}\}$. Likewise $\mathsf{PP}^{-1} \circ \mathsf{PO} = \{\mathsf{PO}\}$, so $N'(d, R) = \{\mathsf{PO}\}$. Again $p(d)$ is necessarily PO since $p(d')$ is PO. The final, most interesting inductive case: d has one child d' with $p(d') = \mathsf{PP}$ and one child d'' with $p(d'') = \mathsf{DR}$. Now $N'(d, d') = \{\mathsf{PP}^{-1}\}$ and $N'(d', R) = \{\mathsf{PP}\}$. $N'(d, R) \subseteq \mathsf{PP}^{-1} \circ \mathsf{PP} = \{\mathsf{PP}, \mathsf{PP}^{-1}, \mathsf{EQ}, \mathsf{PO}\}$. Likewise from d'' we get $N'(d, R) \subseteq \mathsf{PP}^{-1} \circ \mathsf{DR} = \{\mathsf{DR}, \mathsf{PO}, \mathsf{PP}^{-1}\}$. In total, we now know that $N'(d, R)$ is either $\{\mathsf{PO}\}$ or $\{\mathsf{PP}^{-1}\}$. In fact $p(d) = \mathsf{PO}$, because $d < f$ and so cannot be a fountain node. So we want to prove that $N'(d, R) = \{\mathsf{PP}^{-1}\}$ leads to a contradiction. It clearly leads to $N'(e, R) = \mathsf{PP}^{-1}$ for all ancestors e of d, one of which is a child of f - call it f'. For any other child f'' of f, $N'(f'', f') = \{\mathsf{DR}\}$, so $N'(f'', R) \subseteq \mathsf{DR} \circ \mathsf{PP}^{-1} = \{\mathsf{DR}\}$, and thus all leaves $l \leq f''$ must have $N'(l, R) = \mathsf{DR}$, and thus $p(l) = \mathsf{DR}$, so that in fact, appealing to the interpretation I again, R^I is a subset of f''^I and $p(f') = \mathsf{PP}^{-1}$ or EQ, contradicting the fact that f is a minimal fountain. Similarly we get a contradiction from $N'(R, d) = \{\mathsf{PP}\}$, so $N'(d, R)$ and $N'(R, d)$ are $\{\mathsf{PO}\}$ and thus are inverses and $N'(d, R) = \{p(d)\}$, completing the inductive step. □

In other words, what this theorem says is the following: when cell representations of regions are compressed in a certain naive way, which way we have shown to be (near) optimal in Theorem 1, all information about the region in terms of relations to cells can be recovered using a well-established existing technology (path-consistency based constraint solving).

5 Conclusions and Future Work

Some of the results laid out herein may not be entirely surprising. However they lay some of the groundwork for Region Connection Calculi in the context of hierarchical grids and knowledge graphs. We have shown that the naive way of choosing a small number of RCC5 relations to cells so as to represent a region is indeed (close to) optimal, and that this fact is true in a very general setting, regardless of dimensionality, shape, etc. of the hierarchical cells (although it does not apply to some grid systems now in use, such as H3, in which child cells need not be contained in their parents). The methods used to prove this, while not especially deep, are quite general. We established the sufficiency of usual RCC5 constraint reasoning to undo this compression, and its insufficiency for reasoning with grids in general, and provided an alternative method for accomplishing such reasoning.

In addition to compressing the full cell descriptions of single regions in a

vacuum, we may consider some more general types of compression problems:
1. Partial Knowledge. Instead of compressing $\mathsf{desc}(R, I)$ for some variable R and interpretation I, we may have a set of formulas about R that are satisfied by many interpretations J with different $\mathsf{desc}(R, J)$. Can a similar minimal generating set be obtained in this case?
2. Multiple Regions. We may have several region variables R_i and know RCC5 relations between them, not just between the R_i and the cell variables. It may be possible to compress this set of formulas more thoroughly than can be done when we must disregard the relations between regions.
3. More Expressive Logic. While we advocate against using RCC8, there are other more expressive logics that could be worthwhile to reason about spatial data stored by cell representation. For example, replace the qualitative RCC8 relations with quantitative ones, like "d^I is 60% covered by R^I". (This kind of relation was popularized in [2].)

In addition, there is of course also more empirical work to be done to substantiate the added value of our approach in application settings.

Acknowledgement This work was supported by the National Science Foundation (NSF) under award OIA-2033521 "KnowWhereGraph: Enriching and Linking Cross-Domain Knowledge Graphs using Spatially-Explicit AI Technologies."

References

[1] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. MIT Press, third edition, 2009.

[2] M. J. Egenhofer and M. P. Dube. Topological relations from metric refinements. In *Proc. of the 17th ACM SIGSPATIAL International Symposium on Advances in Geographic Information Systems*. 2009.

[3] J. R. et al. S2 geometry library.

[4] Z. K. et al. H3: A hexagonal hierarchical geospatial indexing system.

[5] Z. Gantner, M. Westphal, and S. Woelfl. Gqr – a fast reasoner for binary qualitative constraint calculi. 2008.

[6] P. Hitzler. A review of the semantic web field. *Commun. ACM*, 64(2):76–83, Jan. 2021.

[7] P. Hitzler, M. Krötzsch, and S. Rudolph. *Foundations of Semantic Web Technologies*. Chapman and Hall/CRC, 2010.

[8] A. K. Joshi, P. Hitzler, and G. Dong. Logical linked data compression. In P. Cimiano, Ó. Corcho, V. Presutti, L. Hollink, and S. Rudolph, editors, *The Semantic Web: Semantics and Big Data, 10th International*

Conference, ESWC 2013, Montpellier, France, May 26-30, 2013. Proceedings, volume 7882 of *Lecture Notes in Computer Science*, pages 170–184. Springer, 2013.

[9] J. R. Munkres. *Topology: A First Course*. Prentice-Hall Inc., 1975.

[10] N. F. Noy, Y. Gao, A. Jain, A. Narayanan, A. Patterson, and J. Taylor. Industry-scale knowledge graphs: lessons and challenges. *Commun. ACM*, 62(8):36–43, 2019.

[11] D. A. Randell, Z. Cui, and A. G. Cohn. A spatial logic based on regions and connection. In *Proc. of the 3rd Int. Conf. on Knowledge Representation and Reasoning*. 1992.

[12] J. Renz. A canonical model of the region connection calculus. *Journal of Applied Non-Classical Logics*, 12(3-4):469–494, 2002.

[13] H. Samet. *Foundations of Multidimensional and Metric Data Structures*. The Morgan Kaufmann Series in Computer Graphics and Geometric Modeling. Morgan Kaufmann, 2005.

[14] S. Schockaert and S. Li. Combining rcc5 relations with betweenness information. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*. 2013.

[15] U. Schöning. *Logic for Computer Scientists*. Birkhäuser Boston, 2008.