Qualitative change to 3-valued regions

Matt Duckham[1], John Stell[2], Maria Vasardani[3] and Michael Worboys[3]

[1]Department of Geomatics, University of Melbourne, Victoria, 3010, Australia
[2]School of Computing, University of Leeds, Leeds, LS2 9JT, UK
[3]Department of Spatial Information Science and Engineering, University of Maine, Orono, ME, 04469, USA
[1]matt@duckham.org, [2]j.g.stell@leeds.ac.uk, [3]{mvasardani, worboys}@spatial.maine.edu

Abstract. Regions which evolve over time are a significant aspect of many phenomena in geographic information science. Examples include areas in which a measured value (e.g. temperature, salinity, height, etc.) exceeds some threshold, as well as moving crowds of people or animals. There is already a well-developed theory of change to regions with crisp boundaries. In this paper we develop a formal model of change for more general 3-valued regions. We extend earlier work which used trees to represent the topological configuration of a system of crisp regions, by introducing trees with an additional node clustering operation. One significant application for the work is to the decentralized monitoring of changes to uncertain regions by wireless sensor networks. Decentralized operations required for monitoring qualitative changes to 3-valued regions are determined and the complexity of the resulting algorithms is discussed.

1 Introduction

Imagine a collection of sensors, situated in the plane, measuring values of a single scalar quantity, such as temperature, water salinity, or gas concentration. These measurements may be repeated over an extended time period, thus resulting in time-series at each local sensor node. The overall problem, from which this research stems, is how to construct some high-level qualitative global descriptions of dynamic field of values as it evolves through time.

In the restricted case where the measurand recorded by each sensor takes a Boolean value, we may objectify the dynamic field as a collection of regions of high (or low) intensity that is evolving over time. However, the restriction of the measurand to the Boolean domain provides only a first approximation to qualitative representations of field evolution. A richer approximation would allow the measurand to take one of n values, for some positive integer n. In this paper, we move to the case where the measurand domain is three-valued. This case can model several different kinds of scenarios:

1. The useful model for the domain really is three valued, and may be ordered or unordered.
2. The useful model for the domain is Boolean, but some sensors cannot determine which of the two values applies at a particular moment. This may

occur because either the sensor is measurement uncertainty, is in hibernation mode or not working, or because the domain itself is inherently vague.

For instance, various wireless sensor networks (WSN) have been set for monitoring dynamic spatial fields such as temperature, humidity, and barometric pressure, the combination of which provide indications for potential wildfires [14, 2]. Due to the variability in combinations of the measurands, neighboring sensors may be providing mixed Boolean values. For example, in the network of Figure 1, sensors marked with black filled circles indicate fire, while the ones marked as clear circles indicate safe area. The highlighted area, however, includes readings of both cases and is an area of uncertainty, which cannot be identified either as region on fire or as safe region. The same inherent domain vagueness is met in cases of flood monitoring using water level sensors (e.g., ALERT [3]), or in the detection of toxic gas dissemination using air quality sensors.

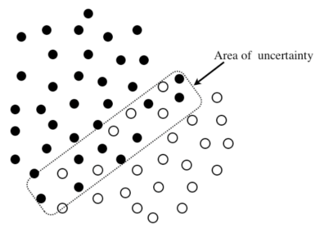


Fig. 1. An area of uncertainty area due to mixed neighboring sensor readings.

The need to abandon the strictly Boolean domain may also result from difficulties in the deployment of a sensor network due to the hostility or scale of the monitored area. In applications such as the joint efforts of Harvard University, the University of New Hampshire, and the University of North Carolina to deploy a WSN to monitor eruptions of Volcan Tungurahua, an active volcano in Ecuador [22] or NASA's VolcanoWeb [23], it may be impossible to cover certain hostile parts of the monitored environment. Sensor networks that serve the study of natural phenomena that intrinsically discourage human presence and spread over large areas, such as hurricanes and tsunamis, may suffer from low density and result in uncertainty in defining the geographic spread of the phenomenon [21].

In different applications, unreliable measurements may be the cause of uncertainty. When offshore wireless sensors are attached to floating buoys (e.g., the CORIE network along Columbia river), the direct light-of-sight is frequently obscured because the height of surface waves exceeds the height of the onshore antennas. The highly dynamic connectivity of such networks may result in uncertain and unreliable sensor readings. Likewise, uncertainty may result from hardware failure in the sensors, which causes coverage holes in the network [1, 21]. Yet a different case is that of responsive WSN—adaptive networks that track salient changes and only keep a certain number of sensors active, leaving the rest of the sensors in hibernation mode [7]—where the dynamic field is monitored by evolution of surface regions that are better represented as fuzzy regions, due to the uncertainty in the extent of the measurand.

For such WSN applications, we may then consider the dynamic field as a three-valued surface, evolving through time, or as a dynamic planar map, where each face takes one of three values. In the latter case, particularly when we are dealing with semantics related to uncertainty, we can think of an evolving collection of regions with broad boundary, which is the approach considered in this work. One method of detecting the topological changes in such regions with broad boundaries would be to communicate data from all nodes to a centralized server, and then process this information using standard tools, like a GIS or a spatial database. However, such centralized approaches are acknowledged to be inefficient, unscalable, and subject to bottlenecks and a single point of failure. The limited bandwidth and node energy resources in untethered networks demand *decentralized* algorithms that are able to detect salient changes in the network itself. In a decentralized algorithm, no single node ever possesses global knowledge of the system state [15]. Therefore, the method presented in this paper is based on a decentralized algorithm approach.

The remainder of this paper is structured as follows: Section 2 provides an outline of the existing literature on detecting changes in crisp or broad boundary regions. In Section 3 we provide the theoretical framework for qualitative change representation, introducing a new tree structure, namely the *BB containment tree*, which allows the depiction of topological events that occur to broad boundary regions. Section 4 provides a decentralized algorithmic approach for monitoring such changes, and Section 5 makes some conclusions.

2 Related Work

This section summarizes the background research on which the work in this paper is based, as well as some related work on the subject of moving vague regions. Previous work, by the authors and others [25, 12, 13] has discussed the case where the measurand is Boolean, providing a formalization and classification of topological changes to regions, as well as a collection of algorithms for topological change detection in a decentralized setting, such as a wireless sensor network. We assume that the regions are bounded by non-intersecting closed curves, embedded in the Euclidean plane. In this case, a collection of regions in the plane is represented as a rooted, directed tree, where the nodes of the tree are the regions. The root of the tree, marked with a double circled node, is the unbounded outer region, and a directed edge exists between two nodes if they share a common boundary. An example is shown in Figure 2.

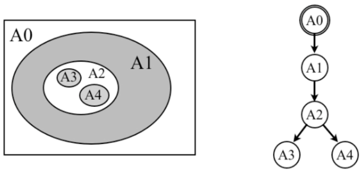


Fig. 2. A collection of regions in the plane and their tree representation.

Topological change to the region collection can be defined by specifying kinds of relations that exist between their corresponding trees. In particular, this formalism can be used to characterize "atomic changes" to the region collection: region insertions and deletions, as well as merges (two kinds—normal and self-merge) and splits (two kinds—normal and self-merge). Geometric properties of region collection evolution may then be represented algebraically, and in some cases this makes the analysis simpler. For example, it is possible to show that there is a normal form for any complex change of a collection of regions as a composition in a specific sequence of atomic changes [13].

While previous research focused on the analysis of types of topological chances that can occur to planar regions through time as they are extracted from the Boolean domain, in this paper focus is on description of the evolution of regions with broad boundaries as they emerge from 3-valued dynamic fields. Such vague regions whose boundaries cannot be easily determined are represented by an egg-yolk pair [4], which is made of two concentric regions. The inner region, referred to as the yolk, represents the parts that definitely belong to the region, while the surrounding region, referred to as the white represents the parts that may or may not belong to the region.

Ibrahim and Tawfik [9, 10] have extended the egg-yolk theory from dealing with spatial regions only, to dealing with regions of space-time. This approach is based on Muller's [17] theory of spatiotemporal regions, which share spatiotemporal, as well as temporal relations. RCC-8 is chosen as the spatial theory combined with Allen's temporal relations and seven motion classes *mc* are defined: *leave, reach, cross, hit, internal, external,* and *split* [17]. Ibrahim and Tawfik [9] redefine the motion classes for vague (egg-yolk) regions. This approach is different than the method of following the evolution of broad boundary regions as the regions themselves are essentially one-dimensional and can only undergo a limited range of qualitative change.

3 Modeling Regional Change

In modeling change it is not sufficient to deal only with starting and ending configurations. For a system of regions in the plane we need to know which regions at the start are related to which regions at the end, and also how this relationship was brought about (by splitting, merging, etc.). To do this with 3-valued regions, it is necessary to extend the tree representation illustrated in the two-type case in Figure 2 (Section 2). When our three types of region are independent of each other, it is relatively straightforward to introduce trees with three types of node and to describe new kinds of splitting and merging based on the types of regions involved. However, when we interpret one type of region as denoting an area of uncertainty surrounding a region (or surrounding a hole in a region), there are some new features with which a formal model needs to contend. In this section we demonstrate how such a formal model can be constructed by introducing additional structure on the trees and how these trees relate to the ones used in the existing literature.

3.1 Varieties of Tripartite Division

We start with a three-way, or tripartite, division of the plane into regions which have boundaries consisting of non-intersecting closed curves. The most general type of such a tripartite division places no limitation on what types of region may appear within each other. This might be interpreted as having three types of substance which do not mix with each other. The dynamics of this situation are then a straightforward extension of the two-type case. Consider one aspect: the splitting of a region. Region splitting may be divided into splits and self-merges. These are distinguished topologically in the plane, as in a split one boundary divides into two boundaries neither of which contains the other, but in a self-merge one boundary divides into two one of which lies inside the other. In the three-region analysis we get six possibilities. This simply provides a combinatorial multiplication of the cases with no structurally significant new features being introduced.

3.1.1 The Intermediate Case

In some applications it is appropriate to regard one type of region as intermediate between the other two. Here it is natural to use black, grey, and white colors in diagrams of the regions, with grey as intermediate between the other two. To ensure grey regions are intermediate means that we require that between any white area and any black one there must be a grey region. This approach might be used to model height of a surface as indicated in Figure 3.

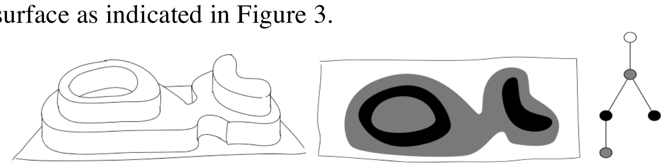
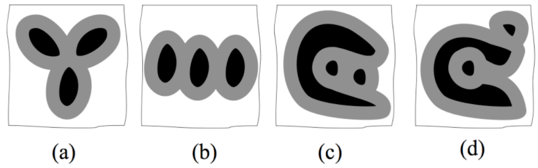


Fig. 3. A schematic landscape, its corresponding height regions and tree.

With this model we can capture notions such as saddle points and plateaus containing both higher and lower areas. Changes to landscape, such as erosion of higher regions to produce level areas and the formation of valleys separating between two relatively higher areas, can be described.

3.1.2 The Broad Boundary Case

A particular case of three types of region is provided by regions of uncertainty. In this case the grey areas are indeterminate regions and the black and white areas are known regions. This type of indeterminate region is the region with a broad boundary, consisting of a certain core and an uncertain area surrounding it. Once we consider the dynamic behavior of such regions we see that it is impossible to be restricted to grey areas which contain a single known (black or white) part. If three such regions merge we can meet cases such as in Figure 4.



(a) (b) (c) (d)

Fig. 4. Topologically equivalent situations within a broad boundary.

The examples shown in Figure 4 are all indistinguishable from a purely topological viewpoint. Each consists of one grey region containing three black regions within it. Informal descriptions of these examples can of course easily distinguish between them. It is possible to see Figure 4(a) as three broad boundary regions which have their broad boundaries fused at one place. In example (b) we can see three broad boundary regions, but this time one is in between the other two. In (c) it is less clear that there are three broad boundary regions, but there is a sense in which the two smaller black regions are inside the larger one. This is not being topologically inside, but is inside in the sense of being inside the convex hull. Finally in (d), only one of the small regions is visually enclosed by the large one, and the other small crisp region is outside.

3.2 Containment Sensitivity

Considering Figure 4(d) we may wish to distinguish three separate regions, but with only one region of uncertainty containing the broad boundaries of all three regions (Fig. 5). This means that in the tree, a grey node no longer stands for a particular region but for a notional broad boundary. To represent this in our tree we need to group these grey nodes together.

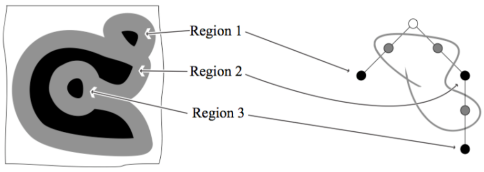


Fig. 5. Notional separation into regions and associated tree.

The next step is a precise description of the kind of clustering the grey nodes shown in Figure 5. Suppose we have a set X which carries a symmetric relation φ. A subset $A \subseteq X$ is said to be φ-connected if given any a, $b \in A$, there is a sequence a_0, a_1, …, a_n of elements of A such that $a = a_0$, $b = a_\mathrm{n}$, and $a_i \; \varphi \; a_{i+1}$ for $i = 0,\ldots, n$ -1. We apply this in the case that φ is the symmetric closure of the adjacency relation in a directed tree. We assume that the three colors of nodes in a tree are grey, black and white; the last two colors used for the 'certain' nodes and the first color for the 'uncertain' ones.

Definition 1 *A **BB containment tree** is a 3-coloured directed tree (T, α) with the constraint that a grey node can be incident with exactly two crisp nodes and where there is an equivalence relation on the set of all grey nodes which is α-connected.*

Given an arrangement of regions in the plane we construct a BB containment tree as follows. Start by constructing a tree of the crisp nodes only. The children of a node k are those nodes n representing regions which lie in the convex hull of k and for which there is no crisp region between n and k. Then augment this tree by adding between every two crisp nodes a grey node and write $m \; \alpha \; n$ if there is an edge from m

to n in the resulting tree. Now impose a relation R on grey nodes, making $g\ R\ g'$ iff there is a node n such that $n\ \alpha\ g'$ and also either $g\ \alpha\ n$ or $n\ \alpha\ g$. Finally make two grey nodes equivalent if the symmetric and transitive closure of R makes them so.

We demonstrate this construction by the example shown in Figure 6. The equivalence classes of grey nodes are indicated by clustering them together in the diagram. The example demonstrates that the structure captured by the tree enables us to distinguish between a region being topologically contained within another and only being in the convex hull of another. A node m represents a region topologically contained in the region represented by n if the grey node between m and n is not to equivalent to the grey node above n.

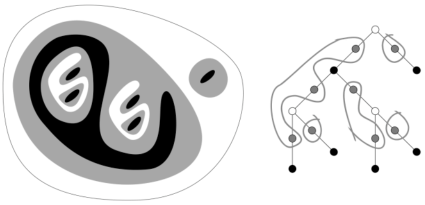


Fig. 6. Regions and the associated BB containment tree.

3.3 Dynamics

Now we consider how to model the dynamic behavior to the broad boundary regions an example of which is provided by Figure 7.

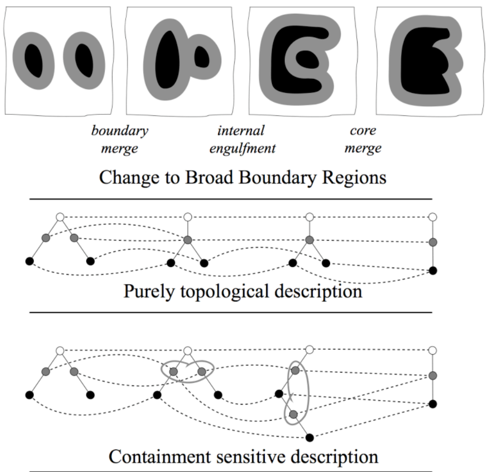


Fig. 7. Dynamics.

Before two BB regions merge into a BB region with a merged core there must be a *boundary merge* as illustrated in the figure. If we model only the topological relationships between the black and grey regions we are unable to detect the stage described as *internal engulfment* in the diagram. This change is modeled by a new type of tree modification in which two certain nodes adjacent to equivalent grey nodes and at the same depth in the tree may move so that one is below the other, while maintaining the equivalence relation on grey nodes.

3.4 Contraction to topology

Although we have seen that the BB containment tree provides a more detailed account of the spatial relationships than is obtained from topology alone, it is important to have a formal justification of the relationship between these two models. One reason is that it is sometimes more appropriate to use the purely topological model and thus being able to derive this from the containment sensitive model avoids the need to maintain two separate structures. Another reason is that by showing how sequences of successive containment sensitive configurations are mapped on to sequences of configurations described only topologically allows us to justify that the new technique is an extension of the existing one. The idea is based on contracting the tree, as shown in Figure 8.

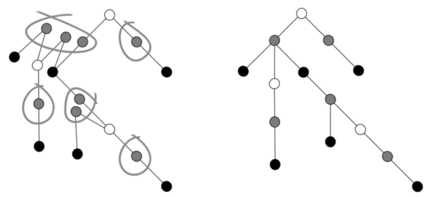


Fig. 8. Contracting the BB containment tree to its topological equivalent.

We form the collapsed tree (T/R, β) as follows. The nodes of T/R are the crisp nodes of T together with the equivalence classes of grey nodes under the relation R. The tree is intermediate—one grey node between any two crisp nodes—so we specify when a crisp node k is adjacent to a gray node G and vice versa there being no other edges (1, 2). In the definition we use α^+ to mean the symmetric transitive closure of α.

$$k \ \beta \ [g] \ \text{ iff } \ k \ \alpha \ [g] \ and \ \neg \exists h \in [g] \ (h \ \alpha^+ \ k) \tag{1}$$

$$[g] \ \beta \ k \ \text{ iff } \ g \ \alpha \ k \tag{2}$$

It does need to be justified that T/R is actually a tree and not merely a graph. To see that this is the case it is sufficient, because of the intermediate nature of the graph, to show that between any two crip nodes in T/R there is a unique path. This can be done by considering operations of expansion and contraction for paths between crisp nodes. The idea is that any path between crisp nodes in T can be contracted to a path between the same nodes in T/R and any path between the nodes in T/R can be expanded to a path in T.

To explain expansion first, any path in T/R will be made up of triples of the form $k\ G\ k'$ where k and k' are crisp and G is a grey node. The triple $k\ G\ k'$ is expanded to the unique path in T between k and k', and by expanding all triples we expand the whole path. Contraction is a reverse process in which the conditions on the equivalence relation ensure that arbitrary paths can be split up into sections which can be individually contracted to triples of this form.

4 Decentralized algorithm

Previous work has already established a number of important algorithmic constructs that can be used as the basis of an extended algorithm for detecting topological changes in regions with broad boundaries. Specifically, two key classes of decentralized algorithms have already been thoroughly explored:

1. Decentralized algorithms for boundary detection and topological change in (crisp) regions; and
2. Decentralized algorithms for detecting the structure of complex areal objects, including (crisp) regions with holes and islands.

The algorithms (described in more detail below) construct decentralized analogs of fundamental centralized data structures and operations: specifically polygonal data structures augmented with boundary orientation (for example, as defined in ISO 19107, [11]), and the semi-line algorithm for point in polygon tests [24]. However, unlike their centralized counterparts, in these decentralized algorithms no single node has access to the global system state. Instead, information is generated, processed, and stored throughout the network itself, in particular at the boundary of the regions themselves.

4.1 Detecting topological changes in crisp regions

Detecting topological changes in crisp regions essentially involves two stages. First, each individual node detects if they are the boundary of a region by querying their immediate one-hop neighbors. Nodes that detect the region using their sensors (e.g., high temperature, low salinity) but are adjacent to nodes that do not detect the phenomenon are adjudged to be at the (crisp) boundary [25].

Second, a higher-level boundary structure is constructed to mediate collaboration across boundary nodes in detecting topological change. For example, [8] and [19] describe algorithms to construct cycles of nodes around the boundary of the region, rather like decentralized "polygons." Both these algorithms rely on nodes being location-aware (having location sensors that determine a node's coordinate location).

One significant difference between these two approaches is that [8] requires communication between nodes across the entire region, where as [19] relies on communication only at the region boundary. This difference has important efficiency implications, since a region containing *r* nodes is expected to have log *r* boundary nodes. (Note that this assumes the boundary measured by the geosensor network is non-fractal; this assumption is reasonable for any granular approximation of the boundary, even in cases where the underlying region is fractal, cf. [5]). Communication complexity (e.g., number of messages communicated as a function of network size) is the overriding computational constraint in resource-limited geosensor networks. Consequently, reducing the number of nodes that must communicate, and so the total number of messages sent and received, increases the efficiency of a decentralized algorithm.

4.2 Detecting the structure of complex sets of regions

Recent work has described an algorithm for detecting the structure of complex sets of regions, and specifically the containment relationships between connected components of these objects [6]. Building on the algorithms above, [6] first computes the orientation of each boundary, using an existing decentralized algorithm for computing the area of a region [20]. Then, a single message from each boundary and marked with an identifier from the origin boundary follows a random walk through the network. At each hop, the message is updated with information about any boundaries it crosses. Using the boundary orientation, a node can locally determine whether the message is crossing into or out of a region. Tracking these changes over the message route enables a node to locally determine when it is at the boundary of a region that contains the boundary from which the message originated. When the containing boundary is crossed, the information about the contained region identifier is stored at the containing region boundary. Having stored this information, the message can then be discarded.

Figure 9 summarizes the structures used in the existing algorithms described above. Boundary nodes (black) detect the region (in gray) and have immediate one-hop communication neighbors (connected by an edge) that do not detect the region. Figure 9 presents only the simplifying case of a network structured as a maximally connected planar graph, where each boundary node neighbors two other boundary nodes, although other network structures are also allowable (cf. [8, 19]). Computing the area of each region component enables determination of consistent boundary orientation (e.g., anticlockwise, *a, b, c, … a* and *x, y, z, … x*). Each boundary node stores only the identity of its next neighbor anticlockwise in the cycle. (Note, dual exterior boundary structures may also be defined, but are omitted from Figure 9 for simplicity).

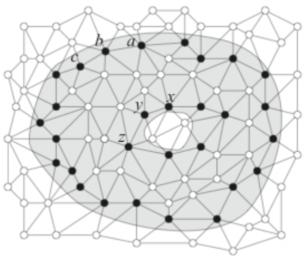


Fig. 9. Boundary structures in a geosensor network monitoring a region with one hole

4.3 Extension to broad boundaries

A natural way to extend the algorithms for crisp boundary detection and tracking to broad boundaries is simultaneously to track the two extremes of the broad boundary (i.e., the interface between the broad boundary and “outside” the region, and the interface between the broad boundary and “inside” the region). To avoid confusion, we refer to these extremes as “interfaces.” Based on the existing algorithms, we can assume that the interfaces for all the regions broad boundaries have been constructed, the orientation has been computed, and the nodes at the interface updated with this information as described above. Figure 10 summarizes the information constructed, alongside with the complete (centralized) containment tree. Note that the information contained in the full containment tree is decentralized, with each interface storing only the identities of its contained regions.

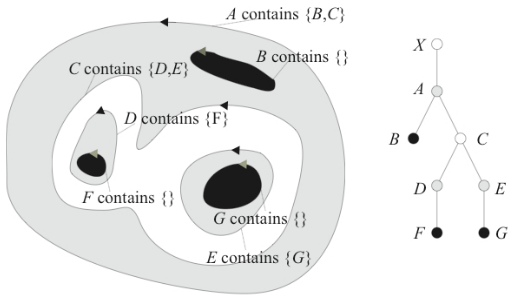


Fig. 10. Interface structure and decentralized containment tree.

Three operations are then required by a decentralized algorithm for monitoring changes to the broad boundary region topology.

1. Routing a message from a contained interface *A* to its unique containing interface (*parent*(*A*)).
2. Adding or removing interface identifiers to the list of contained interfaces (*child*(*A*)) stored at a containing region interface *A*.
3. Discovering and updating the list of contained interfaces, *child*(*A*), of an interface *A*.

In general, operations 1 and 2 above are expected to have communication complexity $O(\log N)$, where N is the number of nodes in the network, since they require only routing along a boundary or other path through the network. By contrast, operation 3 requires routing through an entire region, and so is much less efficient in the worst case leading to $O(N)$ communication complexity.

Table 1 shows how these operations are applied to each of the six atomic topological changes introduced in Section 2. The table provides the name for the change (e.g., appear, disappear, ...); the interfaces that exist before and after the change (e.g., for a disappearance, interface *A* before leads to no interfaces ∅ after the change; for a self merging interface, interface *A* before still exists after the self merge, but has also enveloped a new interface *B*); and the algorithm steps required to maintain the system state (where each boundary node stores a list of the identities of its contained interfaces), based on the three operations above.

As an example, Figure 11 illustrates the operations required to update the decentralized containment tree structure following a split at interface *C*, to form two new interfaces identified *H* and *I*. Note that at the split interface C does not have information about which of the contained interfaces of *C, D* and *E*, are contained which of the two new regions *H* and *I*, necessitating the $O(N)$ discovery operation, flooding throughout *H* and *I* .

Table.1 Decentralized operations required for monitoring qualitative changes to regions with broad boundaries

Change	Before	After	Algorithm steps
Appear	∅	*A*	Route message from *A* to *parent*(*A*) Remove *A* from list of *parent*(*A*) children
Disappear	*A*	∅	Route message from *A* to *parent*(*A*) Add *A* to list of *parent*(*A*) children
Merge	*A, B*	*C*	Update list of *parent*(*C*) children to be *parent*(*A*) ∪ *parent*(*A*) Route message from *C* to *parent*(*C*) Add *C*, remove *A, B* from list of *parent*(*C*)
Self-merge	*A*	*A, B**	Add *B* to list of *A* children Discover and update list of child(*B*) by flooding message through interior *B*
Split	*A*	*B, C*	Route message from *B, C* to *parent*(*A*)** Remove *A*, add *B, C* to list of *parent*(*A*) children Discover and update list of *child*(*B*), *child*(*C*) by flooding message through interiors *B, C*.
Self-split	*A, B**	*A*	Remove *B* from list of *A* children Route message from *A* to *parent*(*A*) Add *child*(*B*) to list of *parent*(*A*) children

* *parent*(*B*) = *A*
** *parent*(*A*) = *parent*(*B*) = *parent*(*C*)

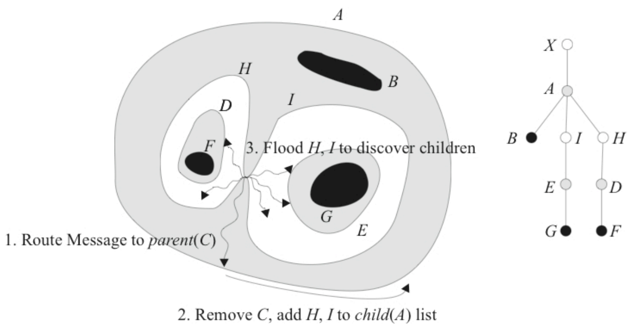


Fig. 11. Example update operations required to monitor splitting of interface *C* (cf. Figure 10).

All the operations used in the algorithm have complexity $O(\log N)$, with the exception of the contained region discovery in self-merge and split, which require $O(N)$ communication complexity. Thus, the efficiency of the algorithm is in the worst case $O(N)$. However, depending on the relative proportions of self-merges and splits that occur to the broad boundary region being monitored, average case complexity may be closer to $O(\log N)$.

5 Conclusions

In this paper we have presented research that extends the theory of dynamic relationships between crisp regions to that of 3-valued regions. The domain examples provided demonstrate the need for this extension to the existing studies of the Boolean case. The approach to modeling such regions goes beyond the purely topological study of relationships between crisp regions and is founded upon the notion of the BB containment tree. We have shown how these trees are a generalization of those used previously by demonstrating that the well-known purely topological description can be obtained by a contraction process.

The second main contribution has been the demonstration of how previously developed decentralized algorithms can be composed to produce algorithms for boundary detection and tracking of regions with broad boundaries. This will be important for further developments of the theory as it provides a means to evaluate theoretical models against the behavior of wireless sensor networks which are an important source of practical examples of regions with broad boundaries.

For future work, we will show that the algorithms are scalable to large datasets by means of simulation experiments. Also, we will develop an account of n-valued dynamic regions for n greater than 3. A third direction would be to model more complex spatial structure within the regions.

References

1. N. Ahmed, S. Kanhere, and S. Ija, (2005) The Holes Problem in Wireless Sensor Networks: A Survey. *Mobile Computing and Communications Review* 9(2): 4-18.
2. A. Antoine-Santoni, J. Santucci, E. de Gentili, and B. Costa, (2007) Using Wireless Sensor Network for Wildfire Detection. A Discrete Event Approach of Environmental Monitoring Tool. In *Proceedings of the First International Symposium on Environmental Identities and Mediterranean Area* (ISEIMA '06), pp. 115-120, IEEE.
3. Automated Local Evaluation in Real-Time—ALERT, www.alertsystems.org (accessed on 01/10/2010).
4. A. Cohn and N. Gotts (1996) The 'Egg-Yolk' Representation of Regions with Indeterminate Boundaries. In P. Burrough and A. Frank (eds) *Geographic Objects with Undetermined Boundaries*, pp. 171-187, Taylor and Francis, Bristol, PA.
5. M. Duckham and J. Drummond (2000) Assessment of error in digital vector data using fractal geometry. *International Journal of Geographical Information Science*, 14(1):67-84.
6. M. Duckham, D. Nussbaum, J-R. Sack, and N. Santoro. Efficient, decentralized computation of the topology of spatial regions. IEEE Transactions on Computers, (Submitted).
7. M. Duckham, S. Nittel, and M. Worboys (2005). Monitoring dynamic spatial fields using responsive geosensor networks. In *ACM-GIS 2005*, Bremen, Germany, pp. 51-60.
8. C. Farah, C.heng Zhong, M. Worboys, and S. Nittel (2008) Detecting topological change using a wireless sensor network. In T. Cova, K. Beard, M. Goodchild, and A.U. Frank, (eds.), *GIScience 2008*, LNCS 5266, pp. 55–69, Berlin, Springer.
9. Z. Ibrahim and A. Tawik (2004) Spatio-temporal Reasoning for Vague Regions. *Canadian AI Conference*: 308-321.

10. Z. Ibrahim and A. Tawik (2005) A Qualitative Spatio-temporal Abstraction of a Disaster Space. In *6th International Symposium on Abstraction, Reformulation and Approximation* (SARA 2005), LNCS 4607, pp. 274-281.
11. ISO. ISO/TC 211/WG 2, ISO/CD 19107 (2003) Geographic information—spatial schema. *Technical report, International Standards Organization.*
12. J. Jiang and M. Worboys (2008) Detecting Basic Topological Changes in Sensor Networks by Local Aggregation. *16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems* (ACM GIS 2008).
13. J. Jiang and M. Worboys, M. (2009) Event-based topology for dynamic planar areal objects. *International Journal of Geographic Information Science*, 23(1), 33-60.
14. Y. Li, Z. Wang, and Y. Song (2006) Wireless Sensor Network Design for Wildfire Monitoring. In *Proceedings of the 6th World Congress on Intelligent Control and Automation* (WCICA 2006), June 21-23, Dalian, China, pp. 109-113, IEEE.
15. N. Lynch. *Distributed Algorithms* (1996) Morgan Kaufmann, San Mateo, CA.
16. A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson (2002) Wireless Sensor Networks for Habitat Monitoring. In *First ACM International Workshop on Wireless Sensor Networks and Applications* (WSNA'02), Atlanta, GA.
17. P.Muller, (2002) Topological Spatio-temporal Reasoning and Representation, *Computational Intelligence* 18(3): 420-450.
18. D. Randell, Z. Cui, and A. Cohn (1992). A Spatial Logic Based on Regions and Connection. In Nebel, B., Rich, C., and Swartout, W. (eds) KR'92. Principles of Knowledge Representation and Reasoning: Proceedings of the Third International Conference, San Mateo, pp. 165-176 Morgan Kaufmann.
19. M. J. Sadeq. In network detection of topological change of regions with a wireless sensor network. PhD thesis, University of Melbourne, 2009.
20. M. J. Sadeq and M. Duckham (2009) Decentralized area computation for spatial regions. In *GIS'09: Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 432–435, New York.
21. M. Umer, L. Kulik, and E. Tanin (2010) Spatial Interpolation in Wireless Sensor Networks: Localized Algorithms for Variogram Modeling and Kriging. *Geoinformatica* 14:101-134.
22. G. Werner-Allen, J. Johnsnon, M. Ruiz, J. Lees, and M. Welsh (2005) Monitoring Volcanic Eruptions with Wireless Sensor Network. In *Proceedings of the second European Workshop on Wireless Sensor Networks* (EWSN'05).
23. Volcano Sensorweb, http://sensorwebs.jpl.nasa.gov/ (accessed on 01/10/2010).
24. M.F. Worboys and M. Duckham (2004) *GIS: A Computing Perspective*. CRC Press, Boca Raton, FL, 2nd edition.
25. M.F. Worboys and M. Duckham (2006). Monitoring qualitative spatiotemporal change for geosensor networks. I*nternational Journal of Geographical Information Science*, 20(10):1087–1108.